End-to-End QoS Provisioning and Traffic Control

신 지 태
성균관대학교 정보통신공학부
jtshin@ece.skku.ac.kr, http://mnet.skku.ac.kr

목 차

- Network Trends
- Network Models
  - Integrated services, RVSP-TE
  - Differentiated services
  - MPLS & DiffServ-Aware MPLS
- QoS Network Function Elements
  - Traffic classification
  - Traffic Conditioning
  - Queue management
  - Scheduling
- QoS Signalling

Current Trends in the Internet

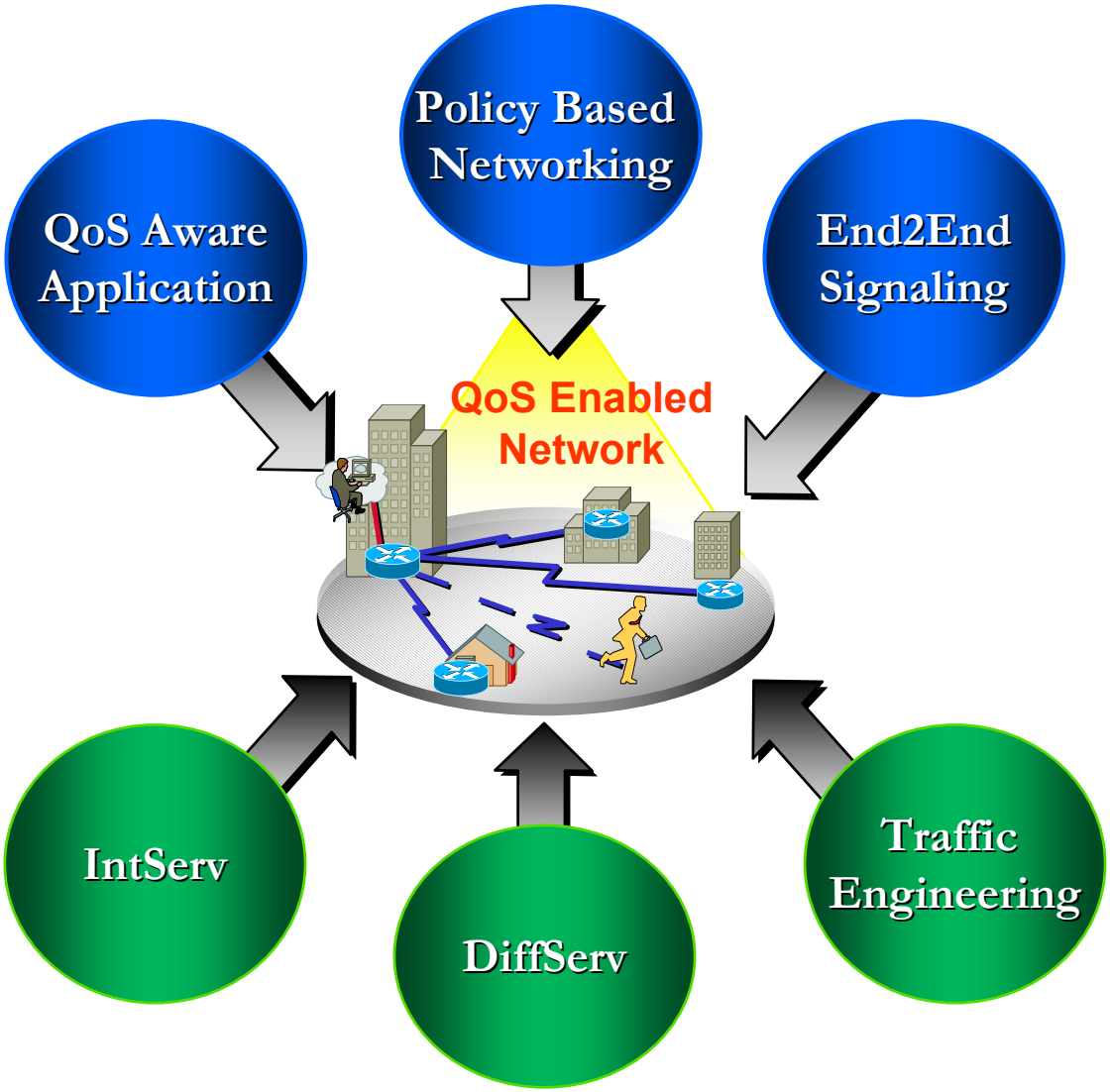
- Diversity of user demand in networking
- Development of real-time Internet applications
- Rapid growth of mobile systems

➡ Something moving towards QoS

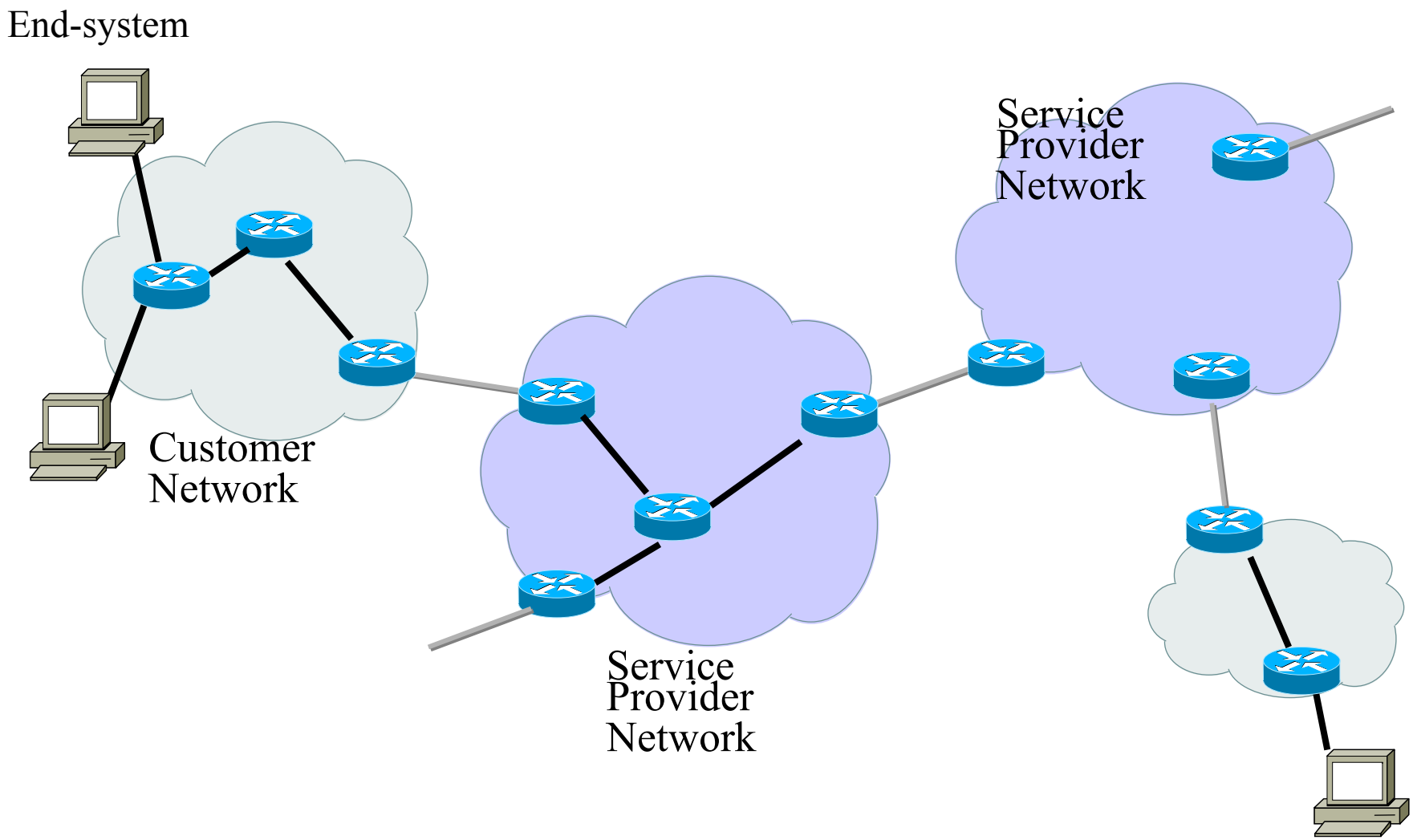
QoS

- the ability to deliver network services according to the parameters specified in a Service Level Agreement.
-a set of capabilities that allow a service provider to prioritize traffic, control bandwidth, and network latency.

Overall Trends

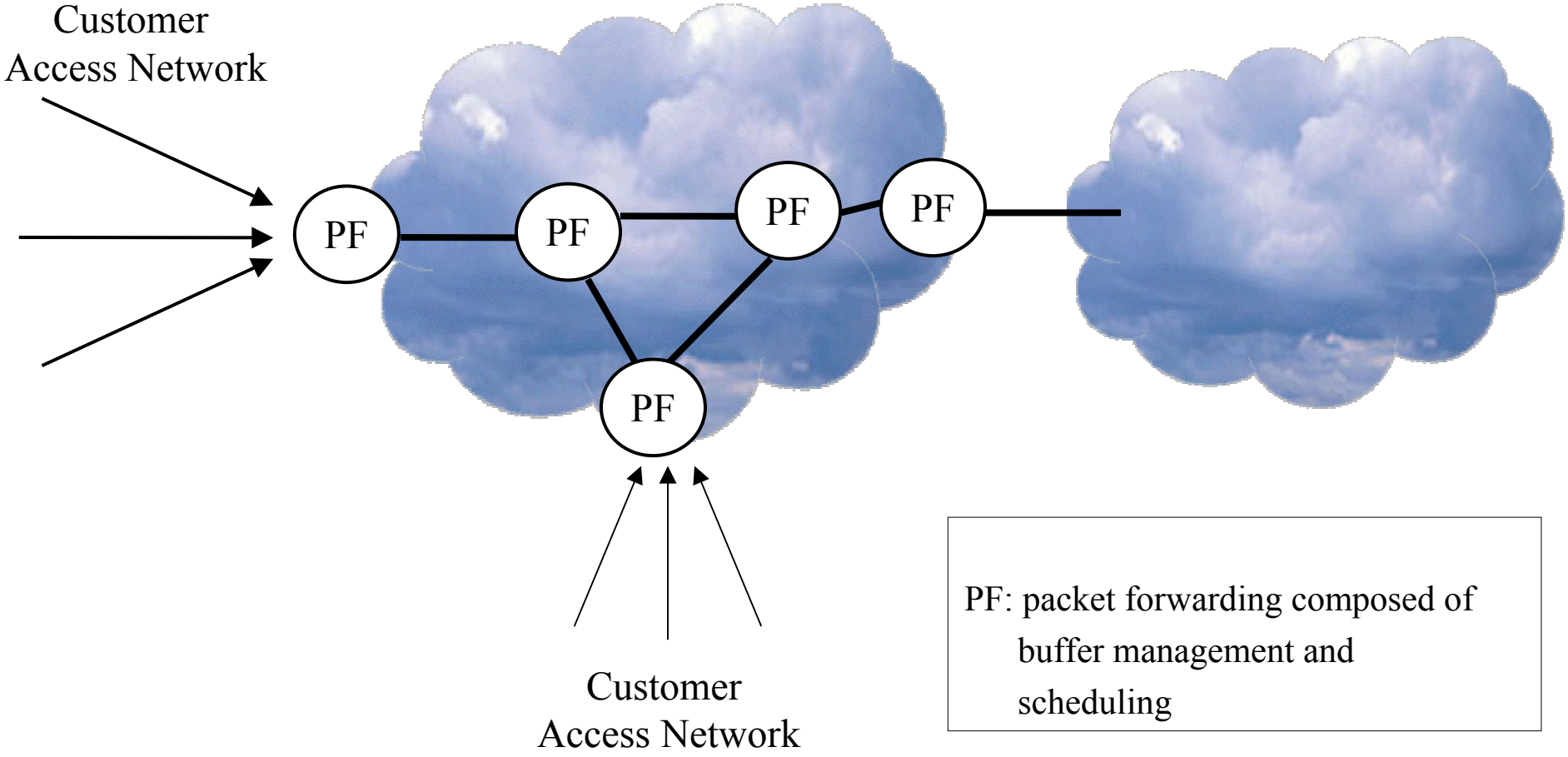


Combined heterogeneous networks



Network

Best-effort of network nodes is less than sufficient!



PF: packet forwarding composed of buffer management and scheduling

More Motivation (1)

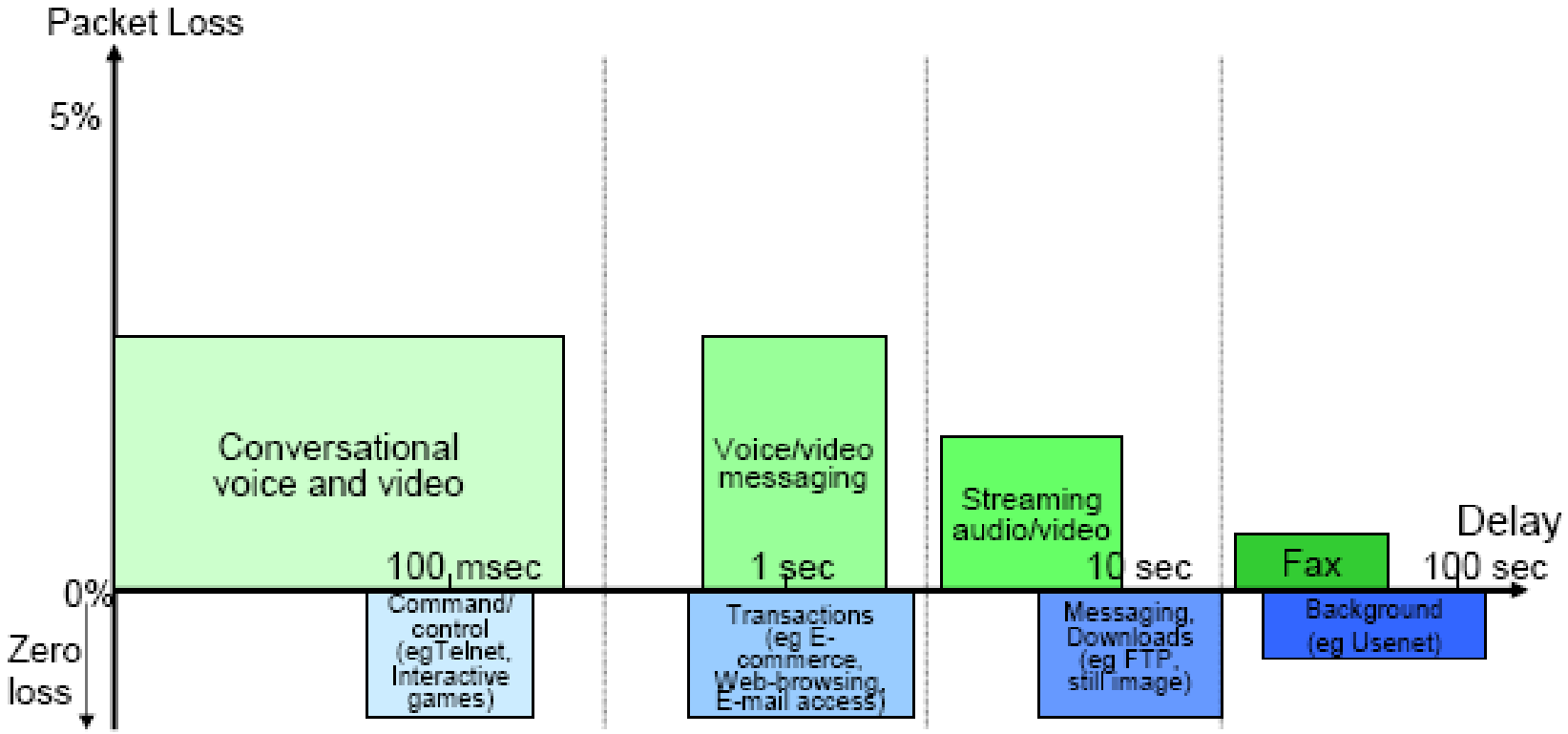
- Emerging real-time streaming applications require more stringent quality requirements.
- One solution : provide abundant bandwidth resource
  - Even massive increase of bandwidth capacity, exponential traffic growth and new bandwidth-demanding applications rapidly consume new bandwidth.
- another promising solution for stringent quality requirements is **QoS** concept
  - Diverse QoS requirements according to applications
    - many traditional TCP-based applications are working well under best-effort network
  - Video requires much higher bandwidth compared to voice and data.

More Motivation (2)

- The benefit of ISP and end-user in QoS enabled network
  - end-user: has a willingness to pay more for higher QoS level, but economical as possible
    - ask a maximum quality with a given budget
  - ISP : can offer(or control) various network services to customer through SLA with different charging mechanism
- Interaction between end-system and QoS enabled network
  - End-system support for QoS provisioning
    - rate adaptation : rate adaptive encoding or frame skip
    - error control : (1) FEC (channel coding, Joint source/channel coding), (2) retransmission, (3) error resilience/concealment
    - source prioritization
  - Network support for QoS provisioning
    - scalable and simple method : e.g., Service Differentiation in aggregate flow

End-User Requirements

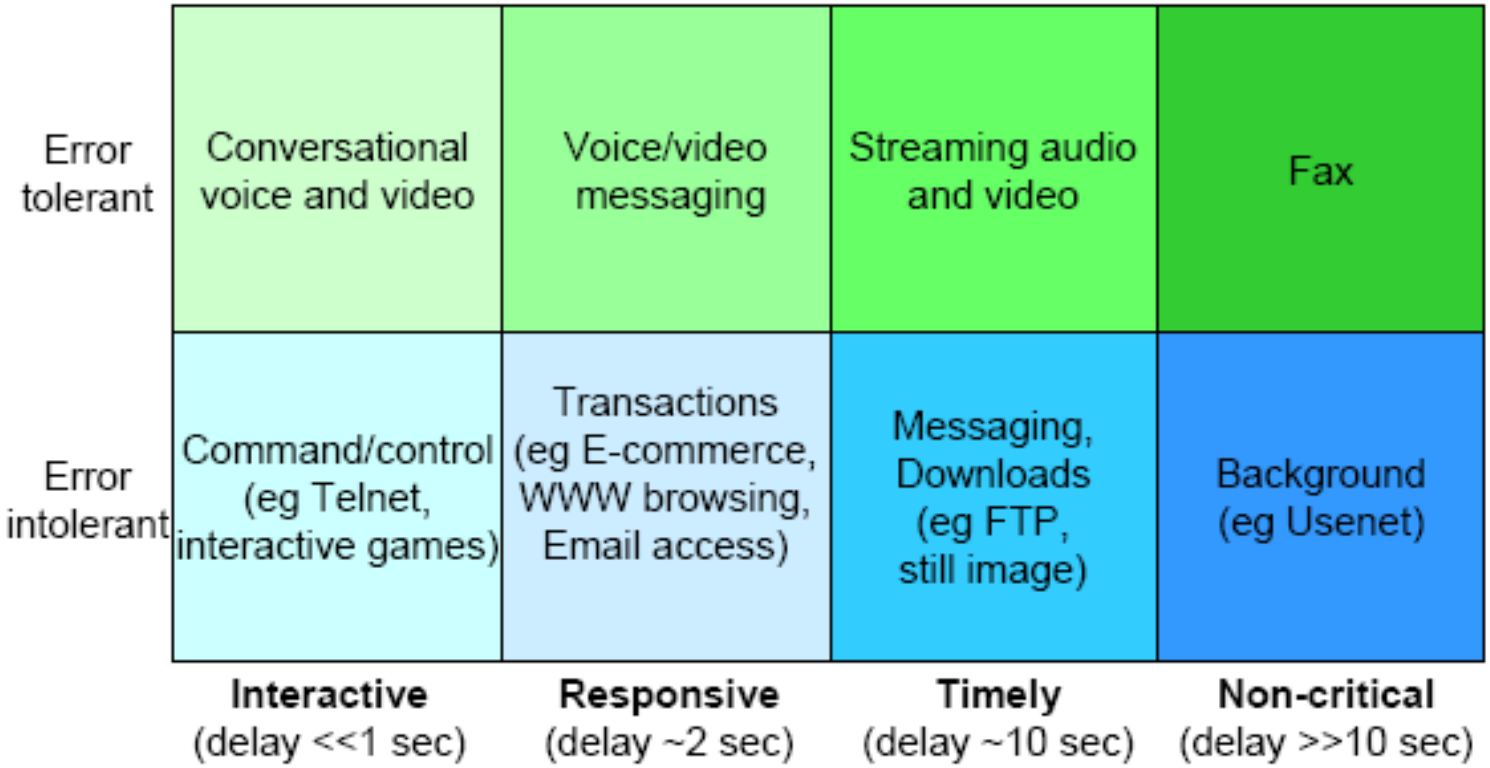
- Wide range of user requirements, depending on application
- QoS is more than just providing performance differentiation within a given service
- No "one-size fits all" solution



From ITU-T Rec. G.1010 (End-user multimedia QoS categories)

End-User QoS Categories

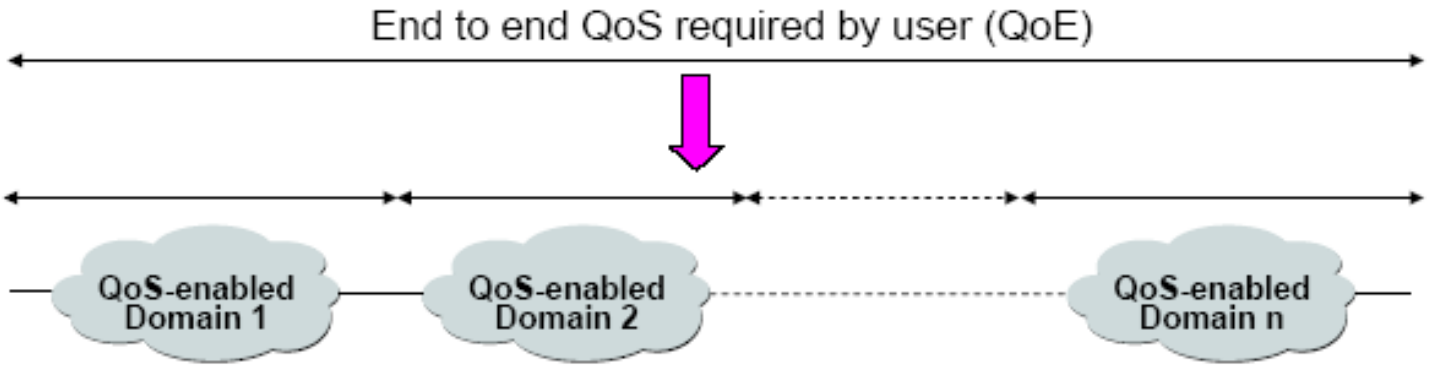
- Group together applications having similar requirements



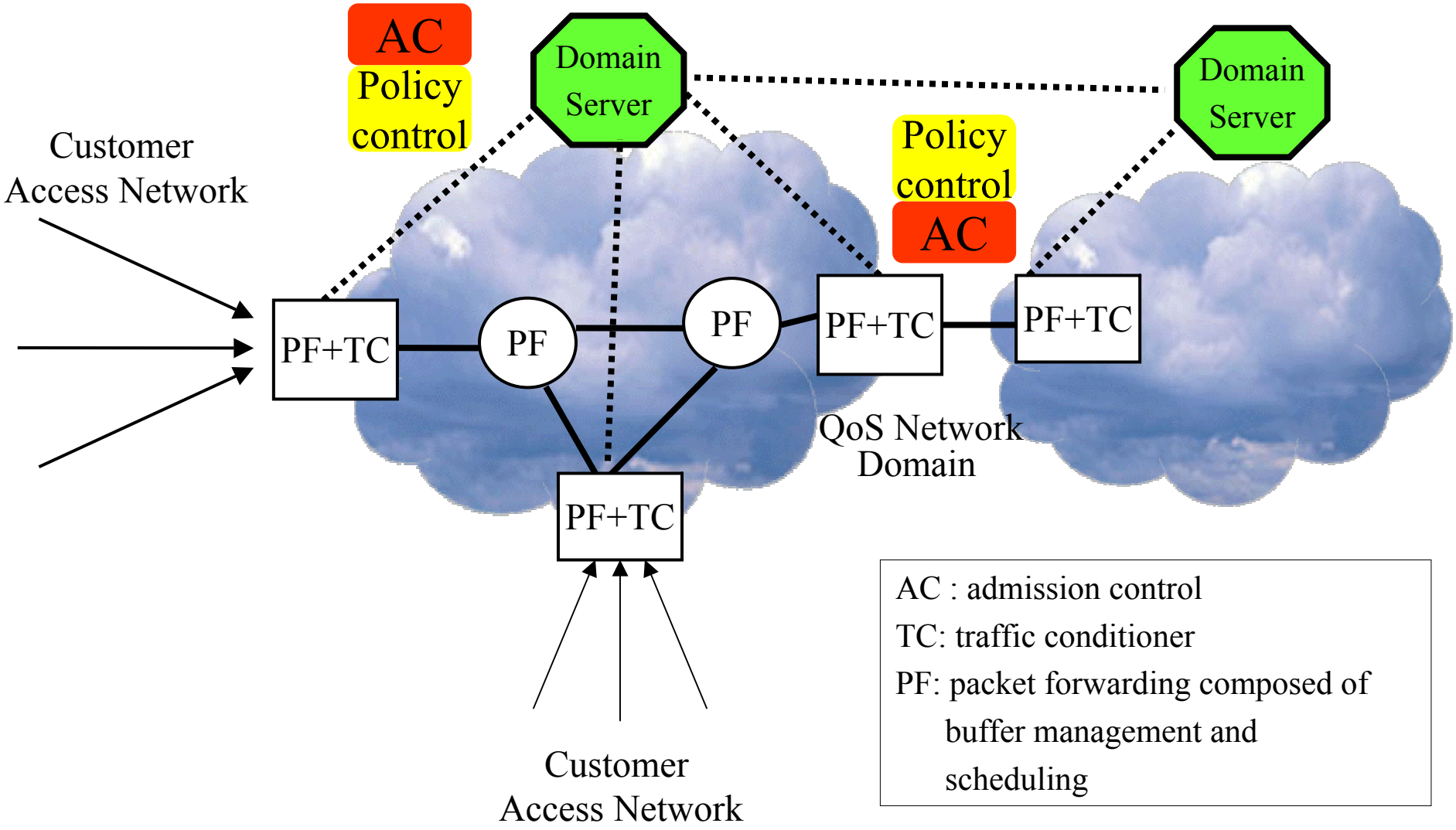
From ITU-T Rec. G.1010

End-to-end QoS

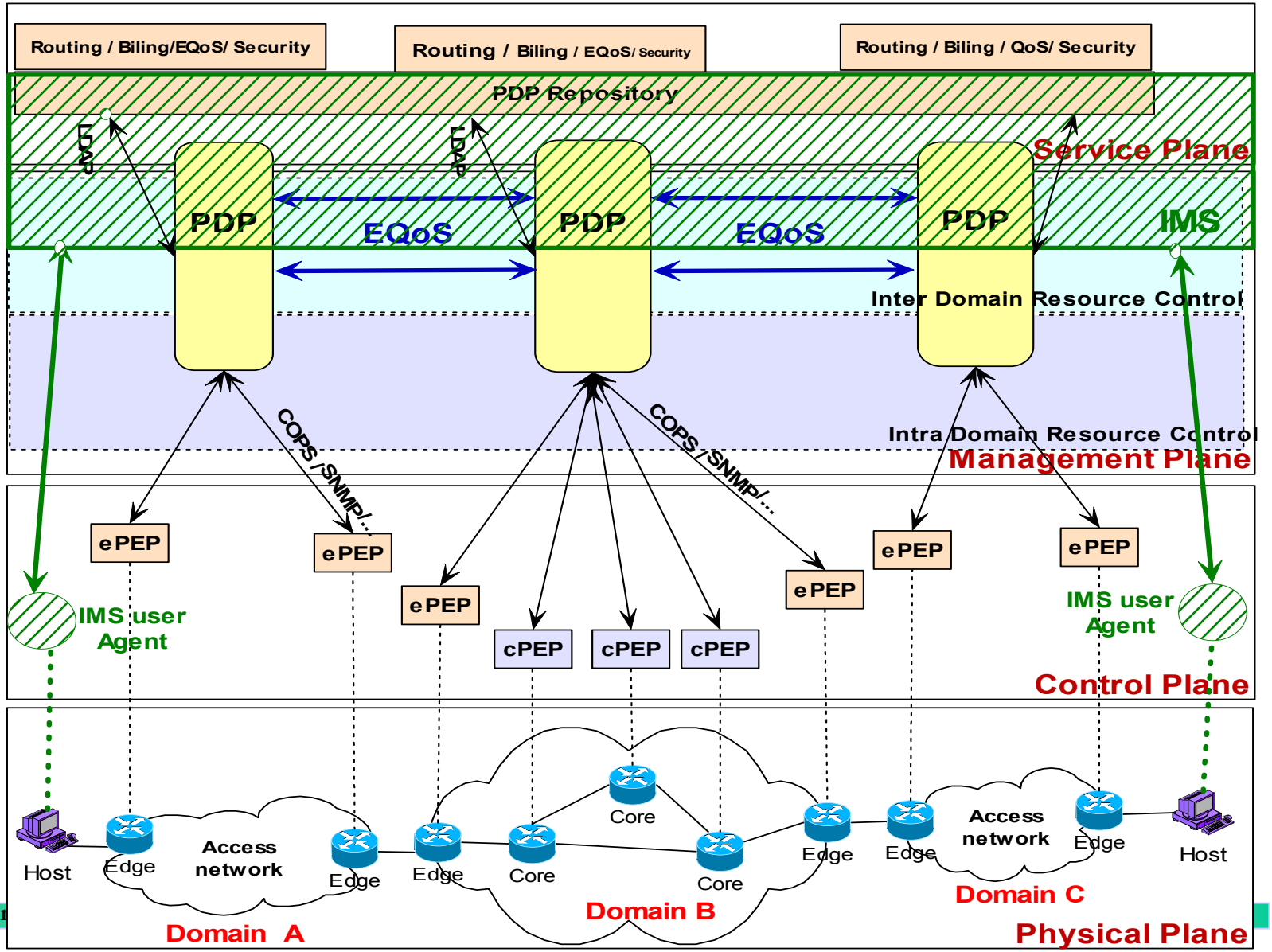
- **User-Driven approach :** Focus on the customer (the one who pays the bill)
- Understand end-user expectations for QoS
  - Quality of Experience (QoE)
  - Real QoS is end-to-end
- Use these to drive requirements for specific QoS mechanisms for individual domains



QoS Network


AC
Policy control
Domain Server
Domain Server
Customer Access Network
Policy control
AC
PF+TC
PF
PF
PF+TC
PF+TC
QoS Network Domain
PF+TC
Customer Access Network
AC : admission control
TC: traffic conditioner
PF: packet forwarding composed of buffer management and scheduling

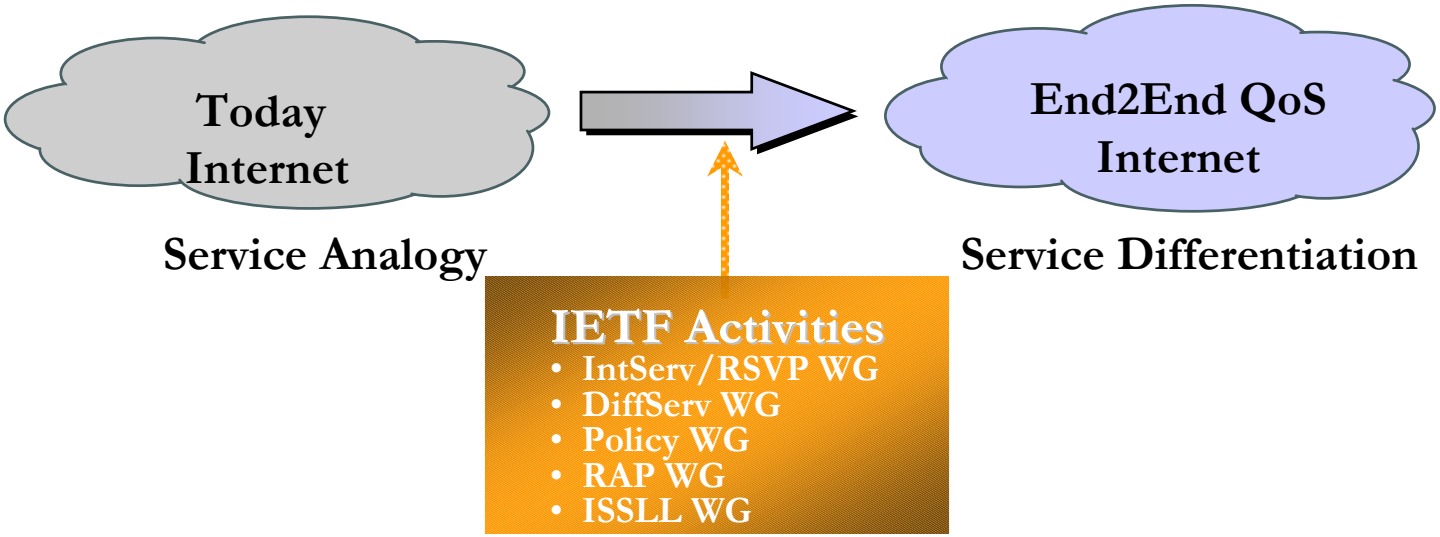
Big Picture in ENTHRONE



Network QoS Models

- Integrated services with RSVP signalling
- Differentiated services
- DiffServ-aware MPLS

History of IETF QoS (1)

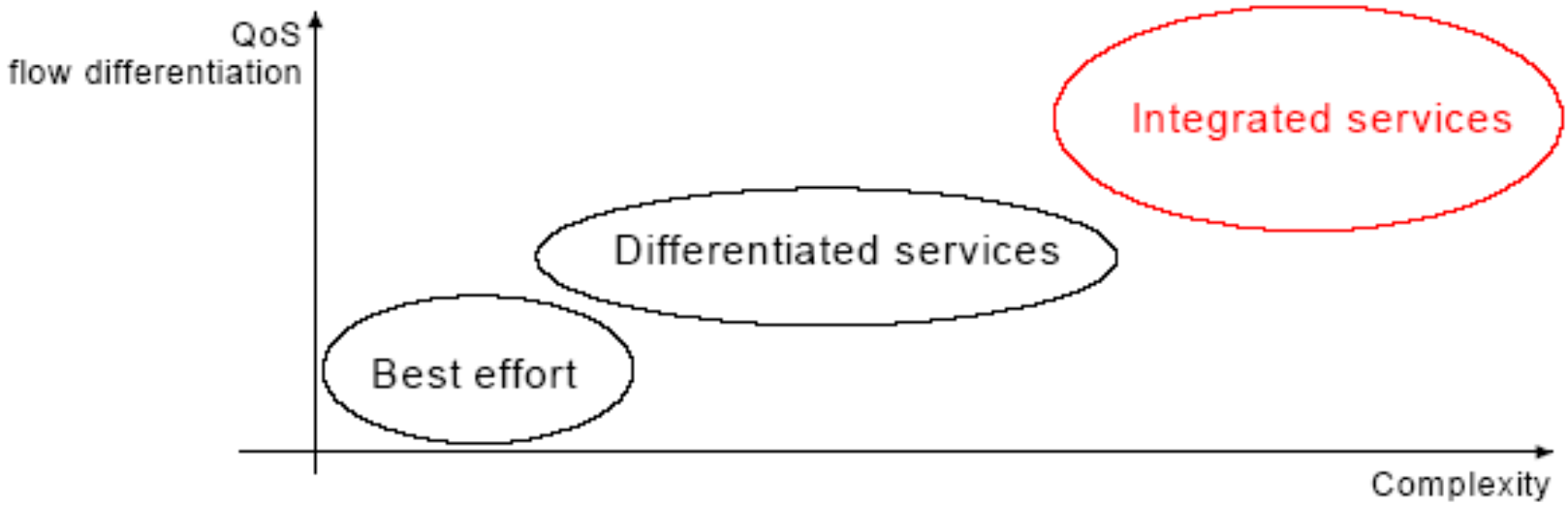


- RSVP: a new resource ReSerVation Protocol, IEEE Network Magazine, Sept 1993
- RSVP working group established, 1994
- Integrated Services working group established, 1994
- Integrated Services over Specific Link Layers working group established, 1996
- RFC 2205 RSVP Functional Specification 1997
- RFC 2208 RSVP Applicability Statement 1997
- RFC 2211 Specification of the Controlled-Load Network Element Service, 1997
- RFC 2212 Specification of Guaranteed Quality of Service, 1997

History of IETF QoS (2)

- A Two-bit Differentiated Services Architecture for the Internet, Internet Draft, 1997
- MPLS working group established, 1998
- Policy working group established, 1998
- Differentiated Services working group established, 1998
- RFC 2381 Interoperation of Controlled-Load Service and Guaranteed Service with ATM, 1998
- RFC 2474 Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers, 1998
- RFC 2475 An Architecture for Differentiated Services, 1998
- RFC 2597 Assured Forwarding PHB Group, 1999
- RFC 2598 An Expedited Forwarding PHB, 1999
- RFC 2689 Providing Integrated Services over Low-bit rate Links, 1999
- IEEE 802.1p released
- RFC 2816 A Framework for Integrated Services Over Shared and Switched IEEE 802 LAN Technologies, 2000
- MPLS Support of Differentiated Services, Internet Draft, 2000
- RFC 2998 A Framework for Integrated Services Operation over Diffserv Networks, 2000

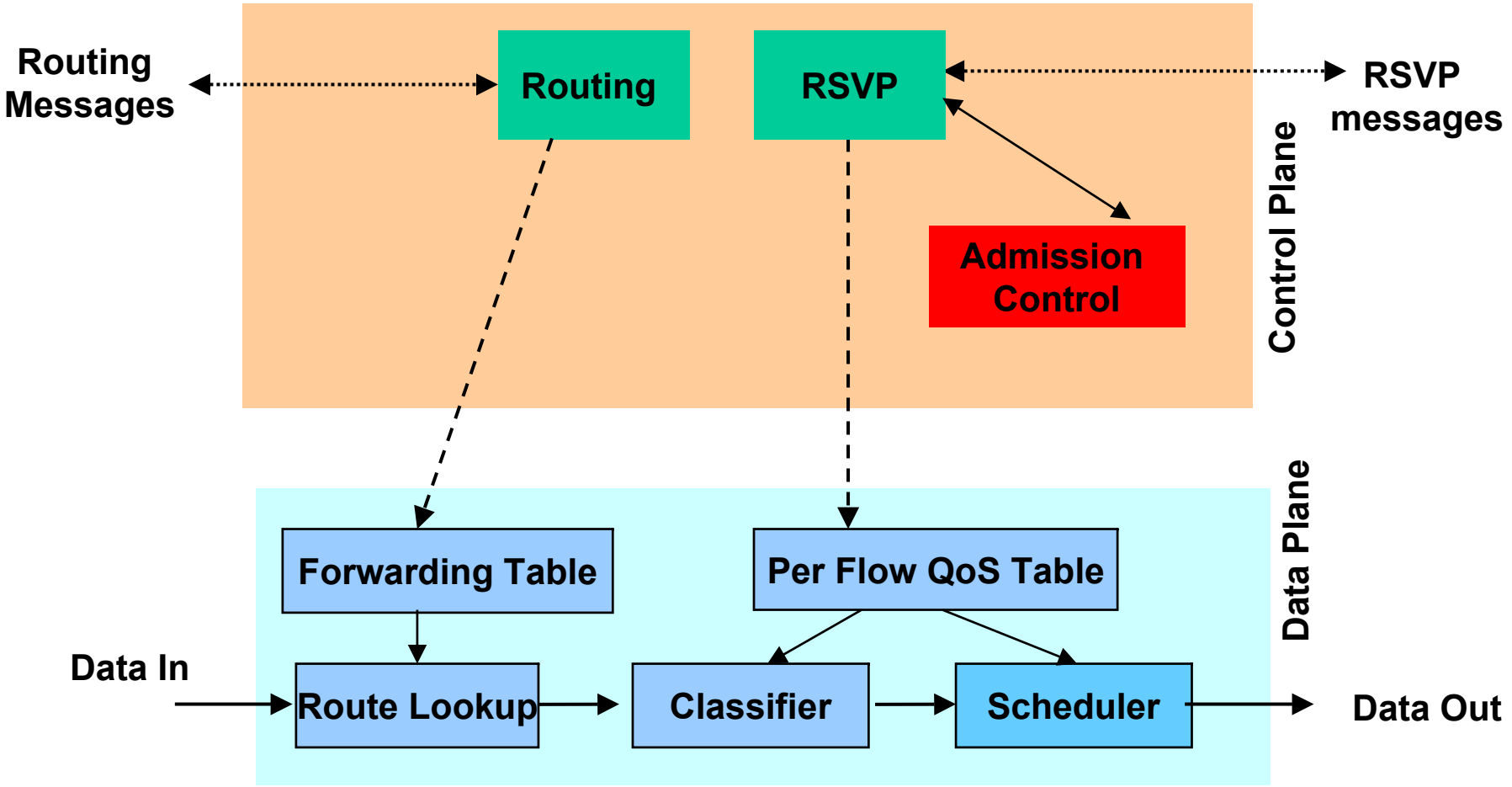
Service scales



Integrated services with RSVP signalling

- what is required to provide a certain of guaranteed service level?
  - Flow differentiation
    - Simple FIFO scheduling will not work!
  - Admission control, Resource reservation : Allocate resources - perform per-flow admission control
  - Flow specification
  - Maintain per-flow state and perform Per-flow classification, Per-flow buffer management, Per-flow scheduling

How Things Fit Together



Service Classes

- Service can be viewed as a contract between network and communication client
  - end-to-end service
  - other service scopes possible
- Three common services
  - best-effort (“elastic” applications)
  - hard real-time (“real-time” applications)
  - soft real-time (“tolerant” applications)

Hard Real Time: Guaranteed Services

- Service contract
  - network to client: guarantee a deterministic upper bound on delay for each packet in a session
  - client to network: the session does not send more than it specifies
- Algorithm support
  - admission control based on worst-case analysis
  - per flow classification/scheduling at routers

Soft Real Time: Controlled Load Service

- Service contract:
  - network to client: similar performance as an unloaded best-effort network
  - client to network: the session does not send more than it specifies
- Algorithm Support
  - admission control based on measurement of aggregates
  - scheduling for aggregate possible

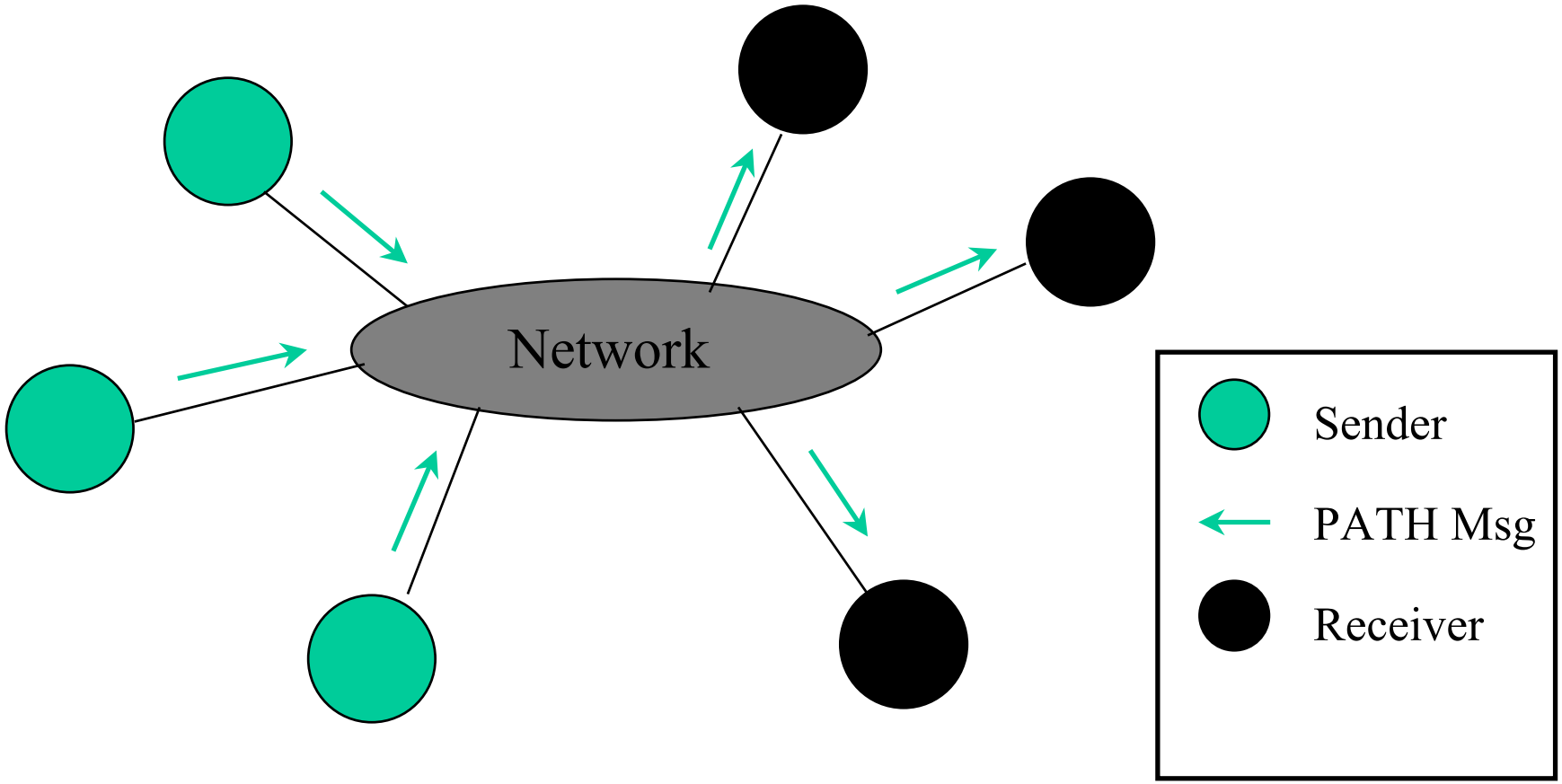
Role of RSVP in the Architecture

- Signaling protocol for establishing per flow state
- Carry resource requests from hosts to routers
- Collect needed information from routers to hosts
- At each hop
  - consults admission control and policy module
  - sets up admission state or informs the requester of the failure

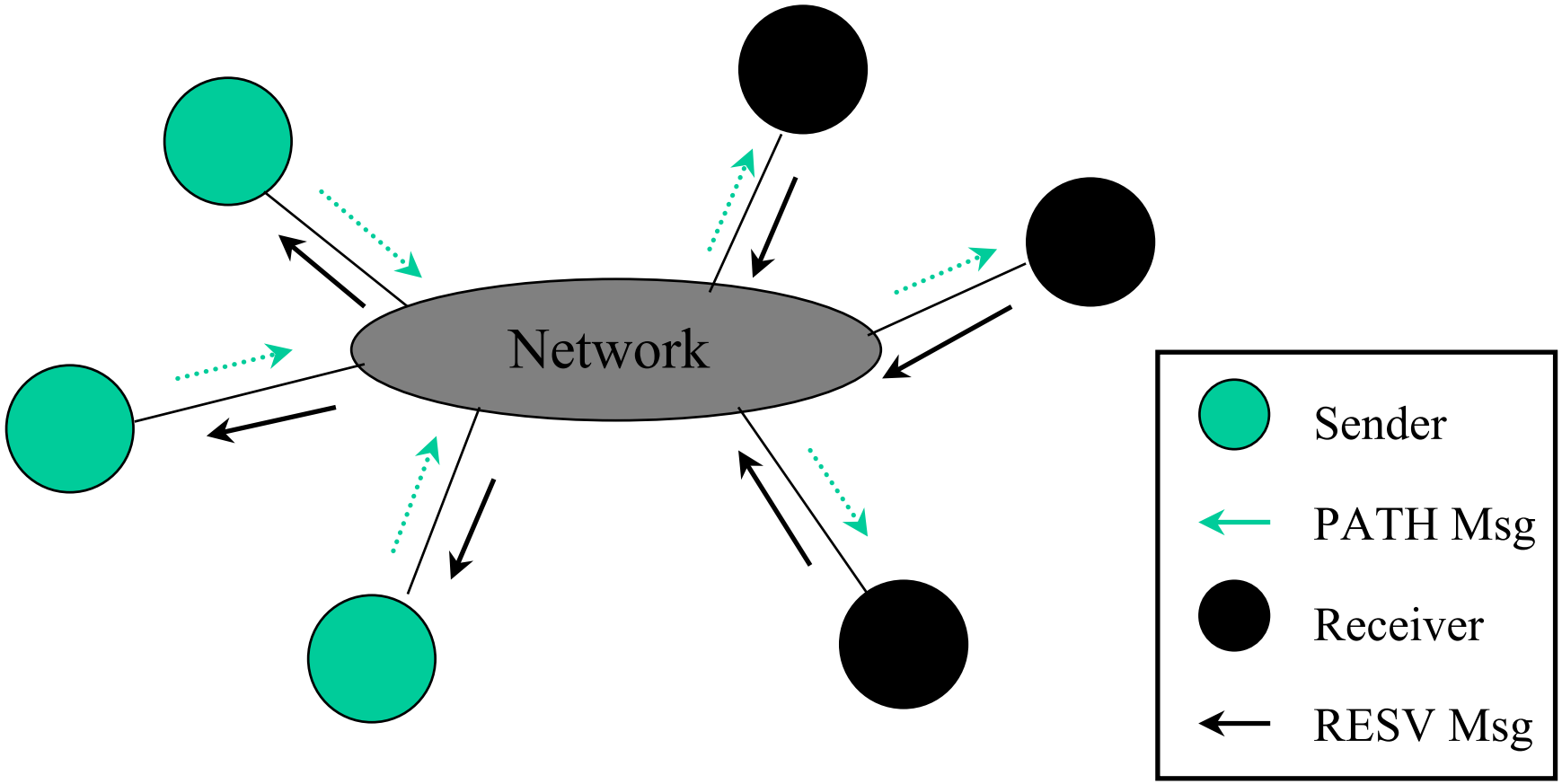
RSVP Reservation Model

- Performs signaling to set up reservation state for a session
- A session is a simplex data flow sent to a unicast or a multicast address, characterized by
  - <IP dest, protocol number, port number>
- Multiple senders and receivers can be in session

The Big Picture



The Big Picture (2)

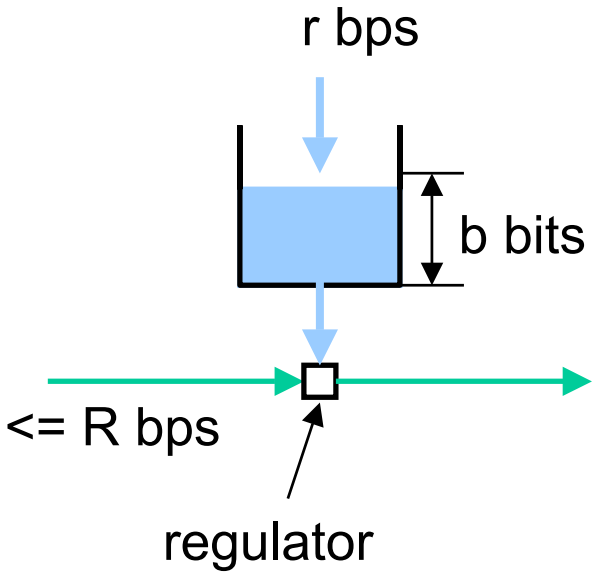


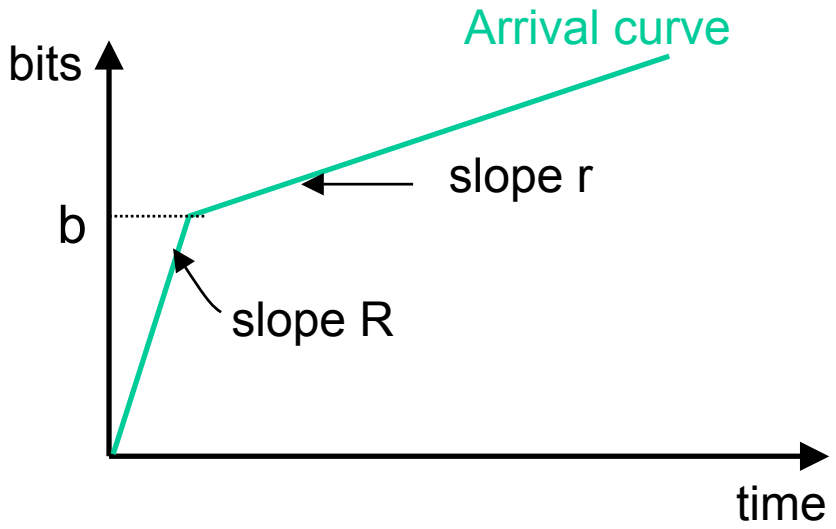
RSVP Basic Operations

- Sender sends PATH message via the data delivery path
  - set up the path state each router including the address of previous hop
- Receiver sends RESV message on the reverse path
  - specifies the reservation style, QoS desired
  - set up the reservation state at each router
- Things to notice
  - receiver initiated reservation
  - decouple the routing from reservation
  - two types of state: path and reservation

Token Bucket

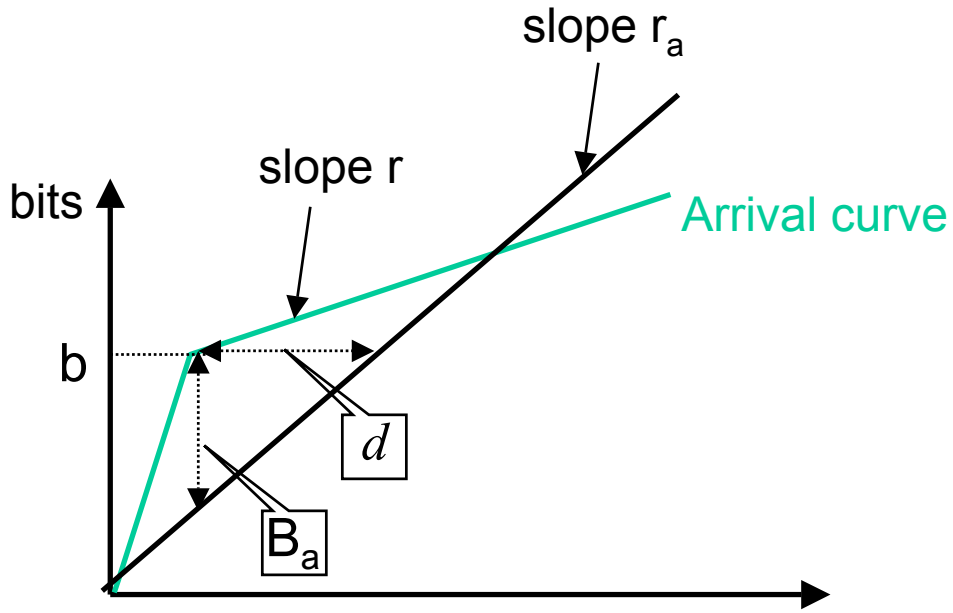
- Characterized by two parameters (r, b)
  - r – average rate
  - b – token depth
- Assume flow arrival rate <= R bps (e.g., R link capacity)
- A bit is transmitted only when there is an available token
- Arrival curve – maximum amount of bits transmitted by time t





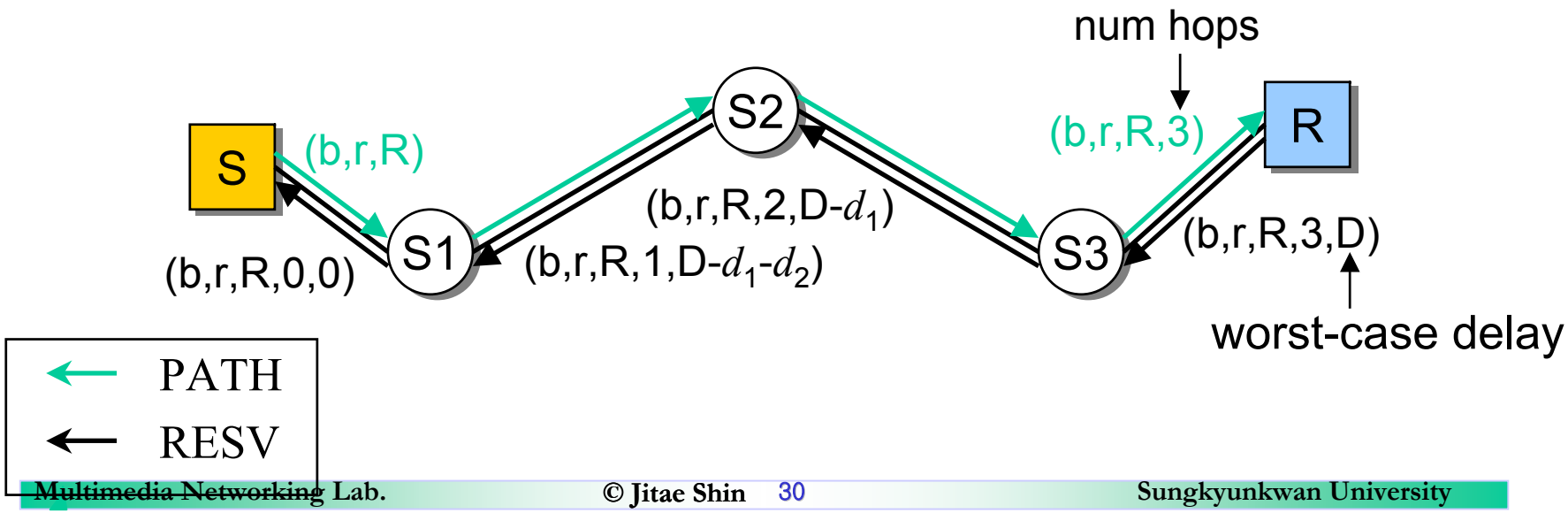
Per-hop Reservation

- Given (b,r,R) and per-hop delay *d*
- Allocate bandwidth r_a and buffer space B_a such that to guarantee *d*



End-to-End Reservation

- When R gets PATH message it knows
  - Traffic characteristics (tspec): (r,b,R)
  - Number of hops
- R sends back this information + worst-case delay in RESV
- Each router along path provide a per-hop delay guarantee and forward RESV with updated info
  - In simplest case routers split the delay



Soft State

- Per session state has a timer associated with it
  - path state, reservation state
- State lost when timer expires
- Sender/Receiver periodically refreshes the state
- Claimed advantages
  - no need to clean up dangling state after failure
  - can tolerate lost signaling packets
    - signaling message need not be reliably transmitted
  - easy to adapt to route changes
- State can be explicitly deleted by a Teardown message

RSVP and Routing

- RSVP designed to work with variety of routing protocols
- Minimal routing service
  - RSVP asks routing how to route a PATH message
- QoS routing
  - RSVP route selection based on QoS parameters
  - granularity of reservation and routing may differ
- Explicit routing
  - Use RSVP to set up routes for reserved traffic

Why did IntServ fail?

- Economic factors
  - Deployment cost vs Benefit
- Is reservation, the right approach?
  - Multicast centric view
- Is per-flow state maintenance an issue?
- What about QoS in general?

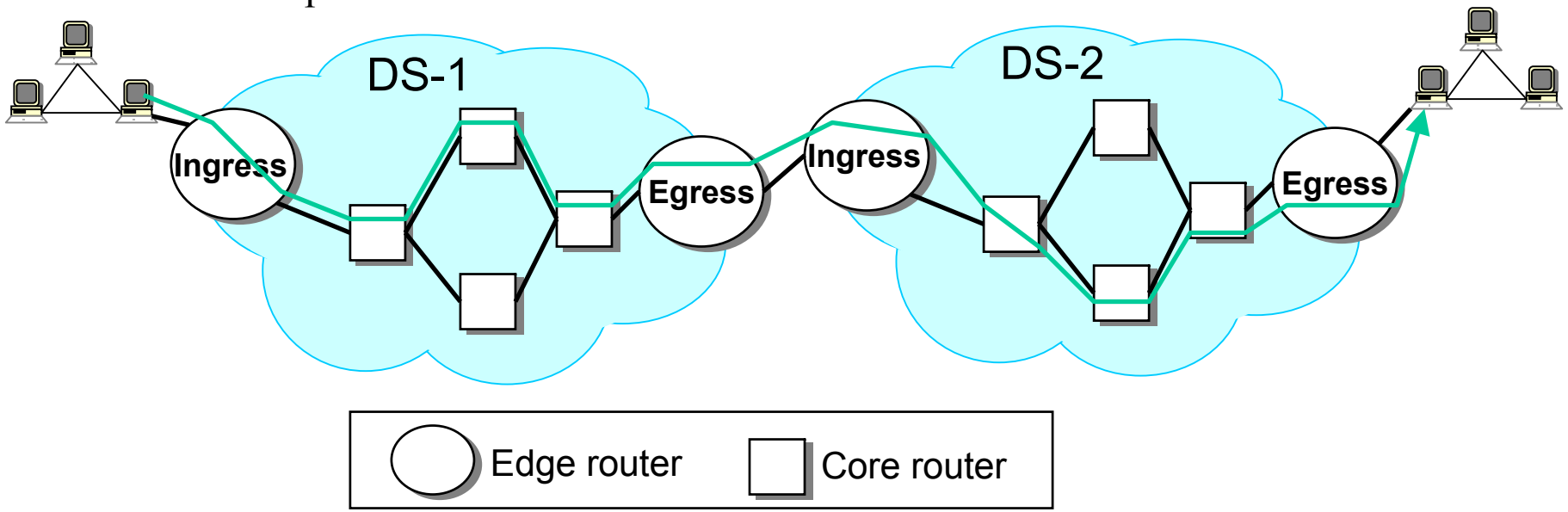
Differentiated Services

- Intended to address the following difficulties with Intserv and RSVP;
- **Scalability**: maintaining states by routers in high speed networks is difficult sue to the very large number of flows
- **Flexible Service Models**: Intserv has only two classes, want to provide more qualitative service classes; want to provide 'relative' service distinction (Platinum, Gold, Silver, ...)
- **Simpler signaling**: (than RSVP) many applications and users may only want to specify a more qualitative notion of service

- The key components of DS:
  - Behavioral Aggregates (BAs)
    - groups of IP flows that are to receive similar forwarding treatment
  - Per Hop Behaviors (PHBs)
    - a description of the forwarding treatment at each network entity
  - Traffic Conditioning (TC)
    - modifies the characteristics of an IP flow to comply with the service limitations
- DiffServ provides only for a differentiation between the relative quality of service experienced by different behavioral aggregates.

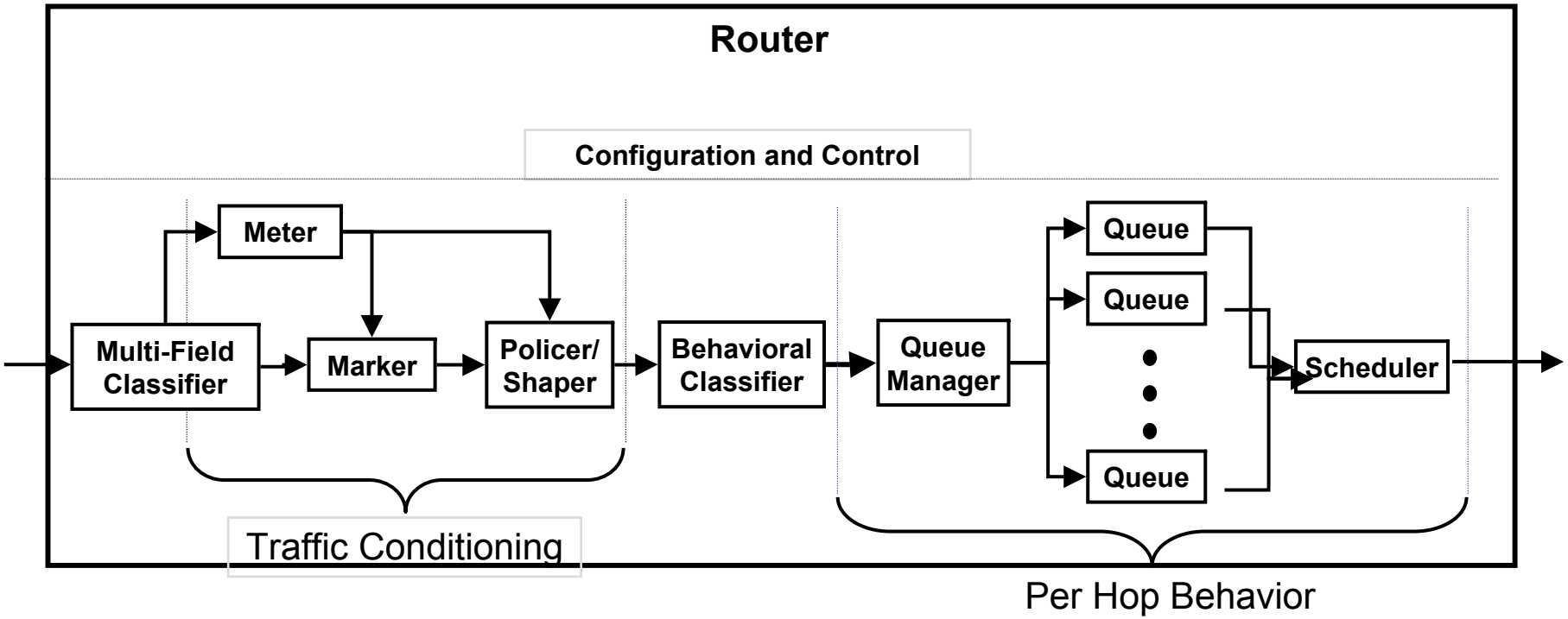
Diffserv Architecture

- Ingress routers
  - Police/shape traffic
  - Set Differentiated Service Code Point (DSCP) in Diffserv (DS) field
- Core routers
  - Implement Per Hop Behavior (PHB) for each DSCP
  - Process packets based on DSCP



DiffServ – PHB and Router

- **Expedited Forwarding (EF) - RFC 2598**: departure rate of packets from a class equals or exceeds a specified rate (logical link with a minimum guaranteed rate), virtual leased line (VLL) type
- **Assured Forwarding (AF) – RFC 2597**: 4 classes, each guaranteed a minimum amount of bandwidth and buffering; each with three drop preference partitions
- Best-effort Forwarding (BF) – RFC 1812



DiffServ – Issues

- Many questions:
  - What are the appropriate queuing disciplines to effect a PHB?
  - What traffic conditioning is most appropriate for each PHB?
  - What are the resulting end-to-end services?
  - How do you support services with strict performance requirements such as voice or video?
  - What happens with routers from different manufacturers in one DS domain?
  - What happens when you mix DS domains?
  - What about DS over different link layers (e.g. wireless)?
  - What about other service quality attributes such as reachability, reliability, data integrity?
  - How do you charge the user for a service that cannot be guaranteed?
- The prevalent answer to most of these questions is to over-provision the network.

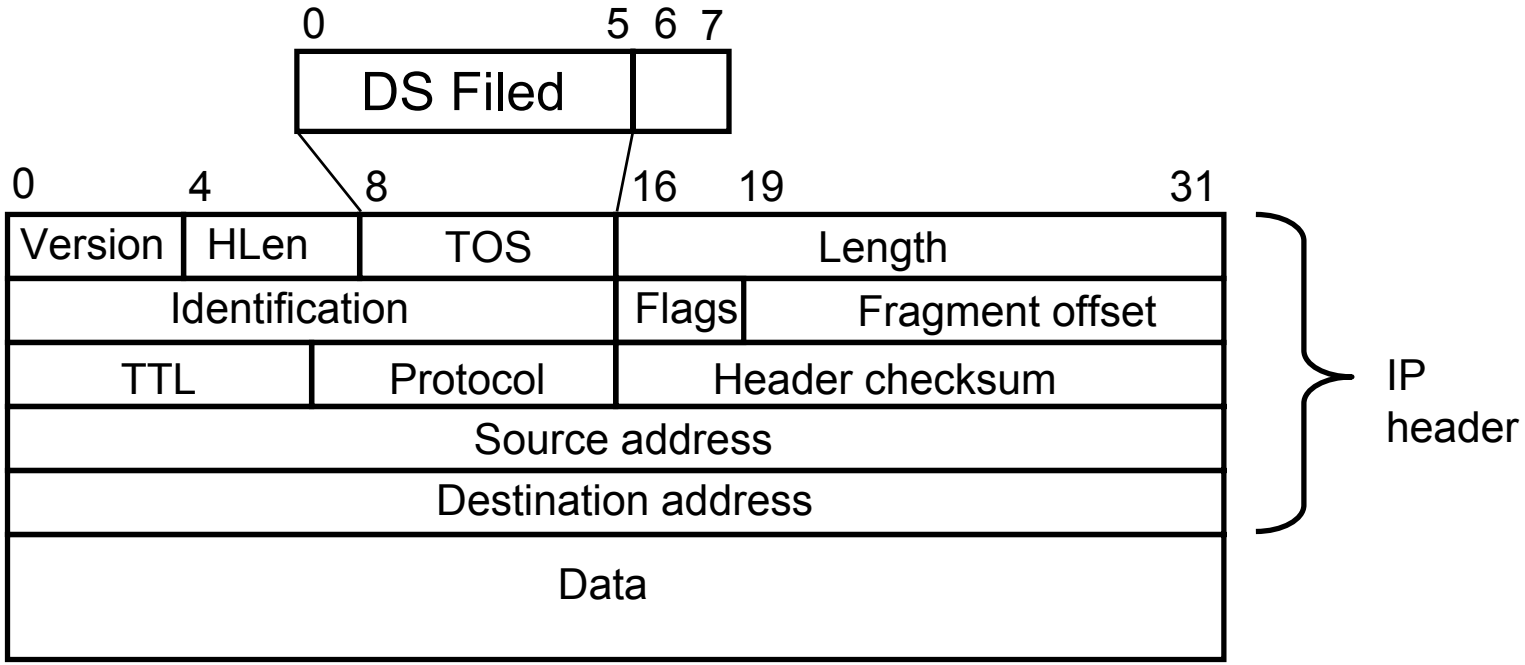
What is the Problem?

- Goal: provide support for wide variety of applications:
  - Interactive TV, IP telephony, on-line gamming (distributed simulations), VPNs, etc
- Problem:
  - Best-effort cannot do it (see previous lecture)
  - Intserv can support all these applications, but
    - Too complex
    - Not scalable

Differentiated Services (Diffserv)

- Build around the concept of domain
- Domain – a contiguous region of network under the same administrative ownership
- Differentiate between edge and core routers
- Edge routers
  - Perform per aggregate shaping or policing
  - Mark packets with a small number of bits; each bit encoding represents a class (subclass)
- Core routers
  - Process packets based on packet marking
- Far more scalable than Intserv, but provides weaker services

Differentiated Service (DS) Field



- DS filed reuse the first 6 bits from the former Type of Service (TOS) byte
- The other two bits are proposed to be used by ECN

Differentiated Services

- Two types of service
  - Assured service
  - Premium service
- Plus, best-effort service

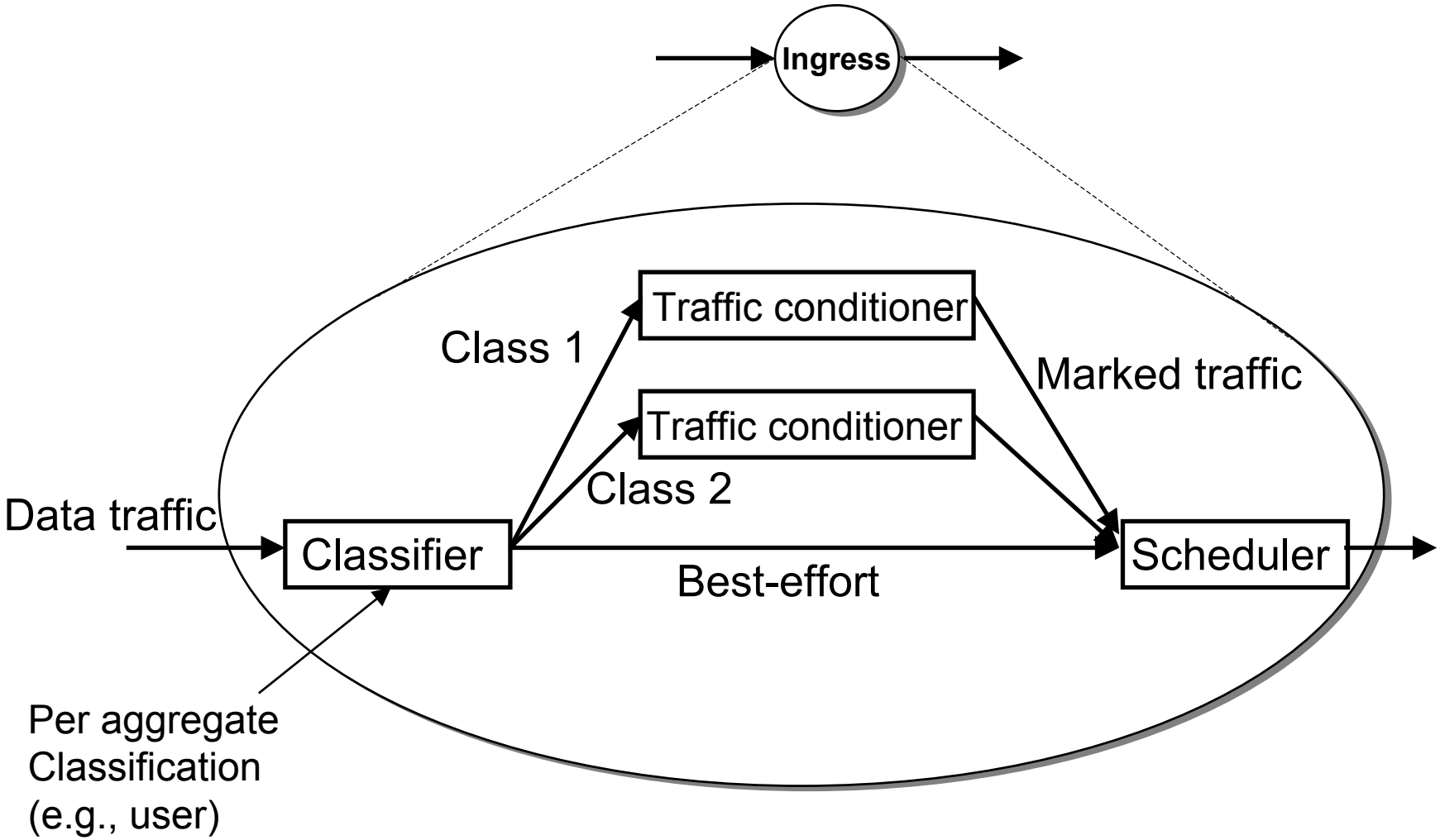
Assured Service [Clark & Wroclawski '97]

- Defined in terms of user profile, how much assured traffic is a user allowed to inject into the network
- Network: provides a lower loss rate than best-effort
  - In case of congestion best-effort packets are dropped first
- User: sends no more assured traffic than its profile
  - If it sends more, the excess traffic is converted to best-effort

Premium Service [Jacobson '97]

- Provides the abstraction of a virtual pipe between an ingress and an egress router
- Network: guarantees that premium packets are not dropped and they experience low delay
- User: does not send more than the size of the pipe
  - If it sends more, excess traffic is delayed, and dropped when buffer overflows

Edge Router

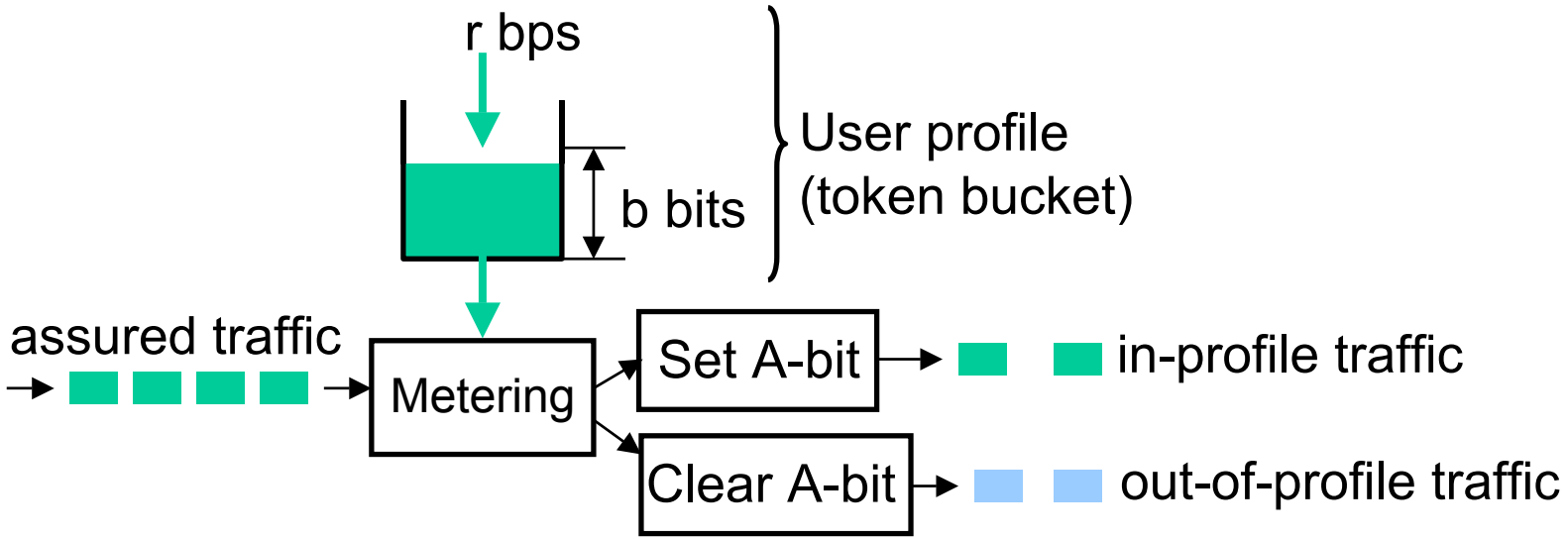

Ingress
Traffic conditioner
Class 1
Marked traffic
Traffic conditioner
Class 2
Data traffic
Classifier
Scheduler
Best-effort
Per aggregate
Classification
(e.g., user)

Assumptions

- Assume two bits
  - P-bit denotes premium traffic
  - A-bit denotes assured traffic
- Traffic conditioner (TC) implement
  - Metering
  - Marking
  - Shaping

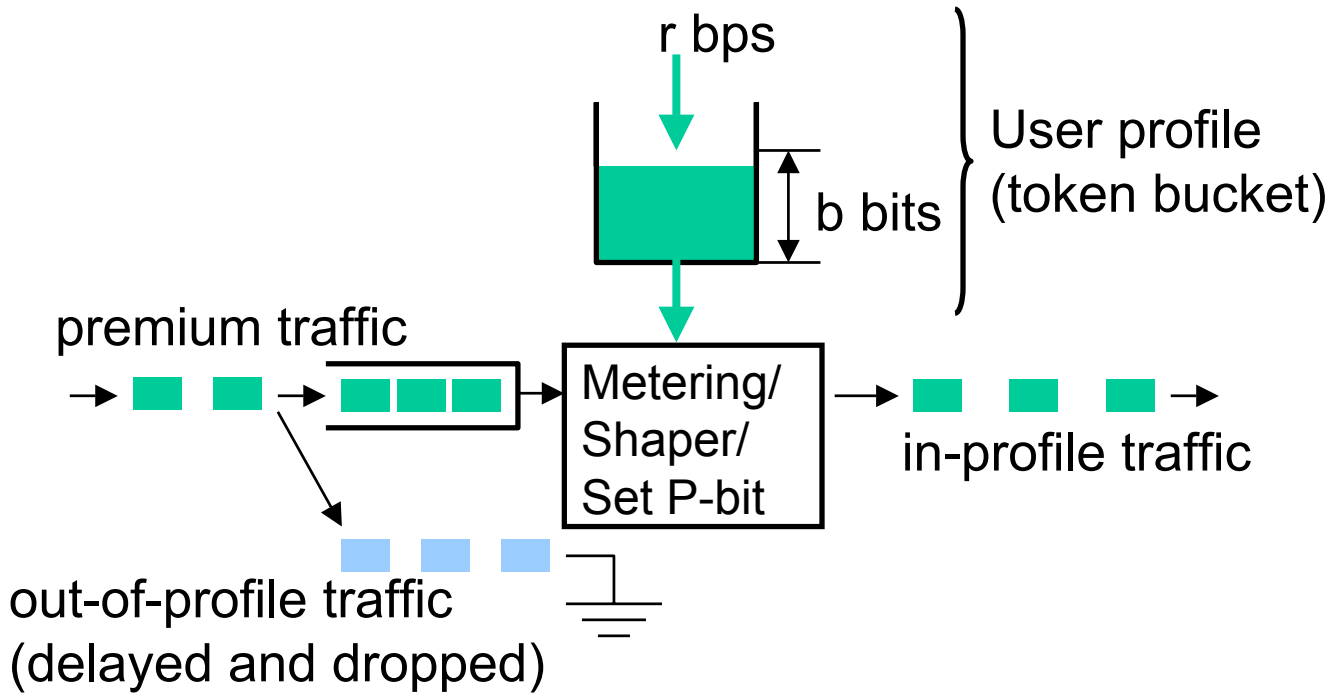
TC Performing Metering/Marking

- Used to implement Assured Service
- In-profile traffic is marked:
  - A-bit is set in every packet
- Out-of-profile (excess) traffic is unmarked
  - A-bit is cleared (if it was previously set) in every packet; this traffic treated as best-effort



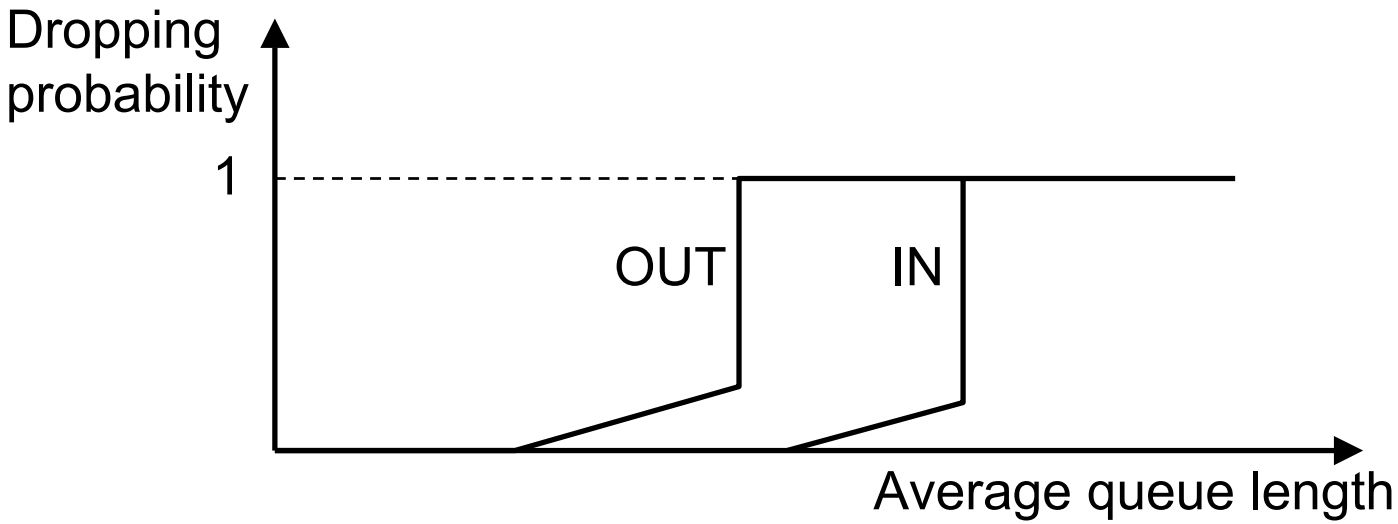
TC Performing Metering/Marking/Shaping

- Used to implement Premium Service
- In-profile traffic marked:
  - Set P-bit in each packet
- Out-of-profile traffic is delayed, and when buffer overflows it is dropped



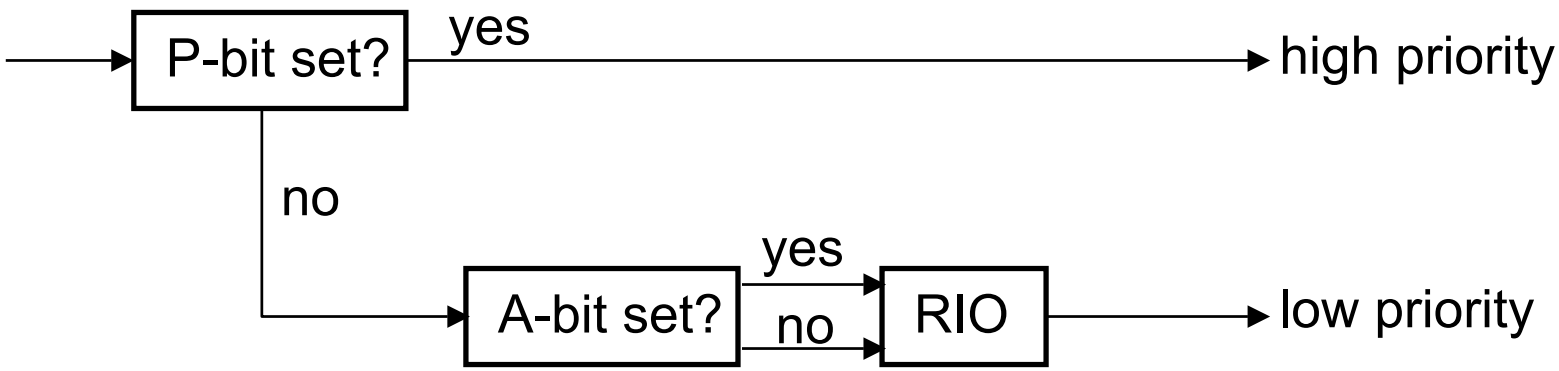
Scheduler

- Employed by both edge and core routers
- For premium service – use strict priority, or weighted fair queuing (WFQ)
- For assured service – use RIO (RED with In and Out)
  - Always drop OUT packets first
    - For OUT measure entire queue
    - For IN measure only in-profile queue



Scheduler Example

- Premium traffic sent at high priority
- Assured and best-effort traffic pass through RIO and then sent at low priority

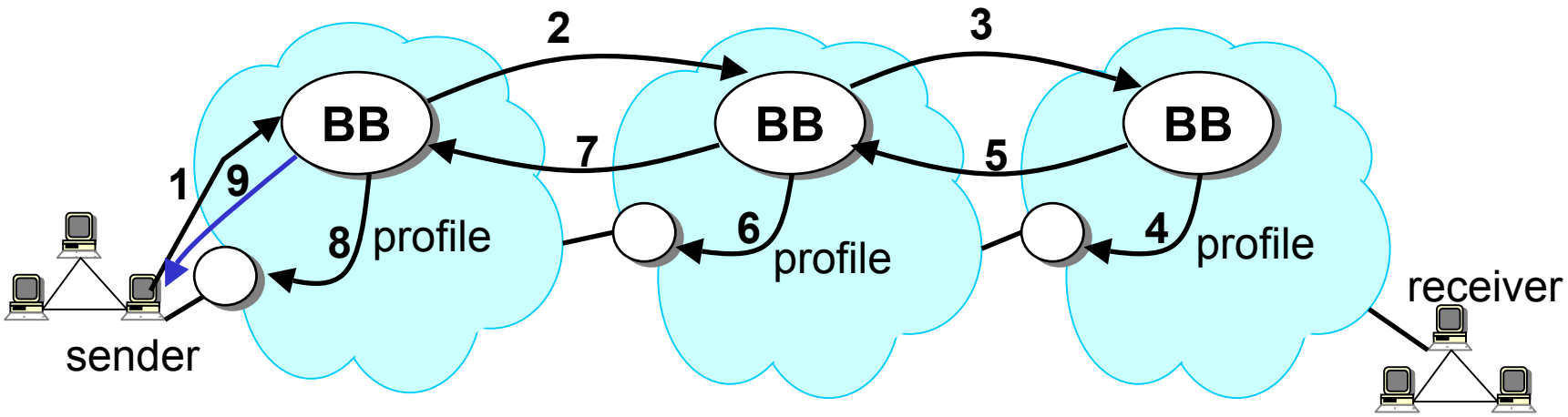


Control Path

- Each domain is assigned a Bandwidth Broker (BB)
  - Usually, used to perform ingress-egress bandwidth allocation
- BB is responsible to perform admission control in the entire domain
- BB not easy to implement
  - Require complete knowledge about domain
  - Single point of failure, may be performance bottleneck
  - Designing BB still a research problem

Example

- Achieve end-to-end bandwidth guarantee



Comparison to Best-Effort and Intserv

	Best-Effort	Diffserv	Intserv
Service	Connectivity No isolation No guarantees	Per aggregate isolation Per aggregate guarantee	Per flow isolation Per flow guarantee
Service scope	End-to-end	Domain	End-to-end
Complexity	No setup	Long term setup	Per flow steup
Scalability	Highly scalable (nodes maintain only routing state)	Scalable (edge routers maintains per aggregate state; core routers per class state)	Not scalable (each router maintains per flow state)

Summary of DiffServ

- Diffserv more scalable than Intserv
  - Edge routers maintain per aggregate state
  - Core routers maintain state only for a few traffic classes
- But, provides weaker services than Intserv, e.g.,
  - Per aggregate bandwidth guarantees (premium service) vs. per flow bandwidth and delay guarantees
- BB is not an entirely solved problem
  - Single point of failure
  - Handle only long term reservations (hours, days)

IntServ vs DiffServ Techniques

Features / Network Technique	Adavantages	Disadvantages
IntServ	- Standardized E2E signalling protocol (RSVP) - Requested services offered (CL and GS)	- RSVP not scalable and complex due to need for core routers to maintain control state for each flow - Common agreeemnt among all ISP's needed for E2Econnections (on RSVP use) - Not suitable for E2E QoS
DiffServ	- Scalable (focusing on traffic aggregates with similar service requirements) - Requested services offered	- No signalling protocol to check the availability of resources E2E - Not suitable for E2E QoS
IntServ in the access & DiffServ in the core	- Scalable (DiffServ) - Standardized IntServ/RSVP signalling protocol for inter-domain	- Common agreement needed between ISP's on the use of IntServ/RSVP for inter-domain signalling

IntServ/RSVP vs. DiffServ (1)

IntServ/RSVP

- Per flow based QoS
- Be suitable for small size network
- Explicit set-up mechanism
- Fitting well for a policy framework

DiffServ

- Per aggregate based QoS
- Be suitable for backbone network
- No signaling
- Simple core

How about combining the merits of both models?

In order to provide scalable end2end QoS, RSVP/IntServ and DiffServ models can be used as complementary technologies in the access and the core networks respectively.

IntServ/RSVP vs. DiffServ (2)

- Reservation based strategies can provide more varied QoS than feedback-based schemes
- Will this be the end of TCP? Highly unlikely. Applications are established, heterogeneous networks, etc.
- Diffserv is middle ground: no intelligence v.s. high intelligence with Intserv

- Use of RSVP Request QoS in a Differentiated Services network?
  - RSVP is originated by the end system (sender or receiver) as if the end system were talking to an Integrated Services network
  - The DS network traps the RSVP messages at the network edge, and maps the RSVP QoS request into a DSCP
  - The DSCP is encapsulated into an object - the DCLASS object and returned to the sender as part of the RESV message payload
  - The sender can then mark each packet with the appropriate DSCP
  - Diff Edge can be used to effect Admission Control in a DS access network.
  - RSVP DClass is supported by the Winsock 2 API (Windows 2000)

Integrated Model (1)

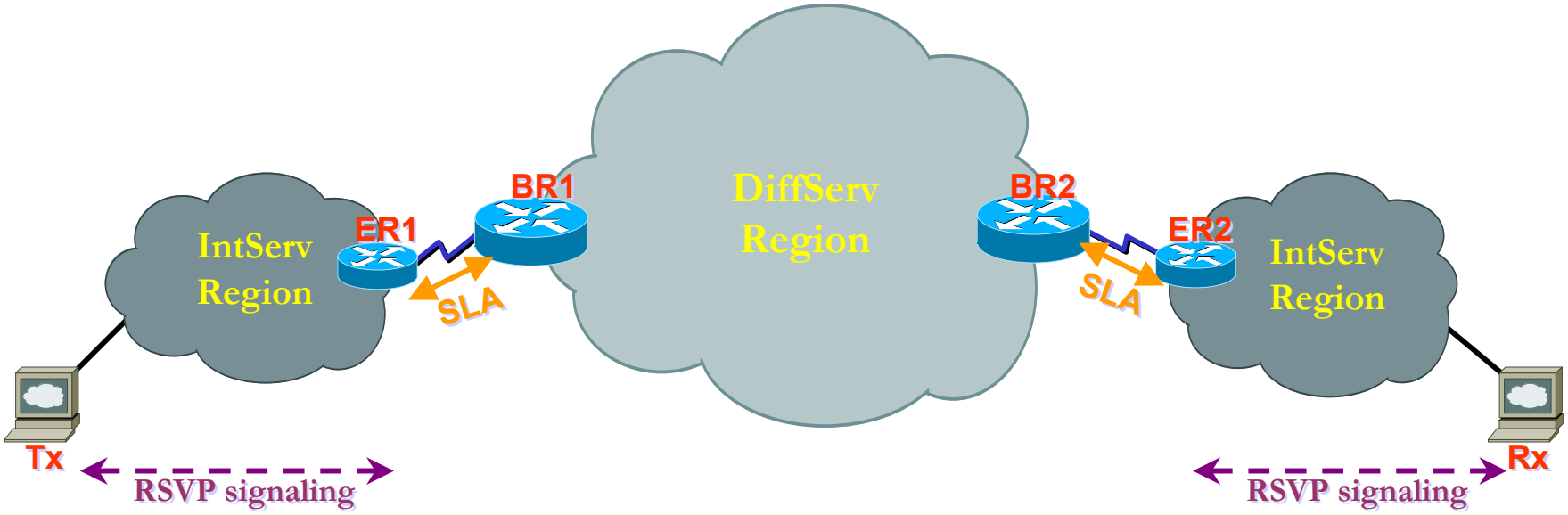
Integrated Framework Building Scenarios

Various scenarios exist for resource management in the DiffServ region to meet the needs of end2end IntServ flows

- Resources statically provisioned by static contracted SLA
- Resources dynamically provisioned by RSVP
- Resources dynamically provisioned by BB (e.g., COPS/DS)

Integrated Model (2)

Static provisioning model

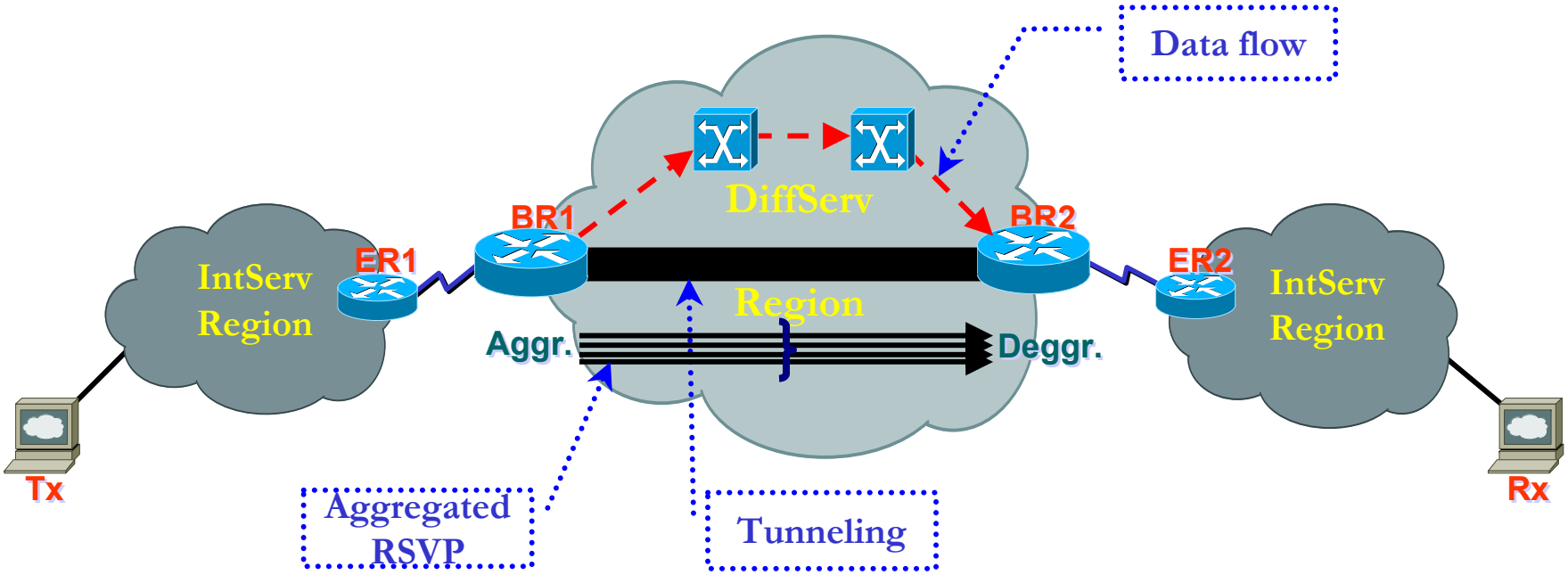


Integrated Model (3)

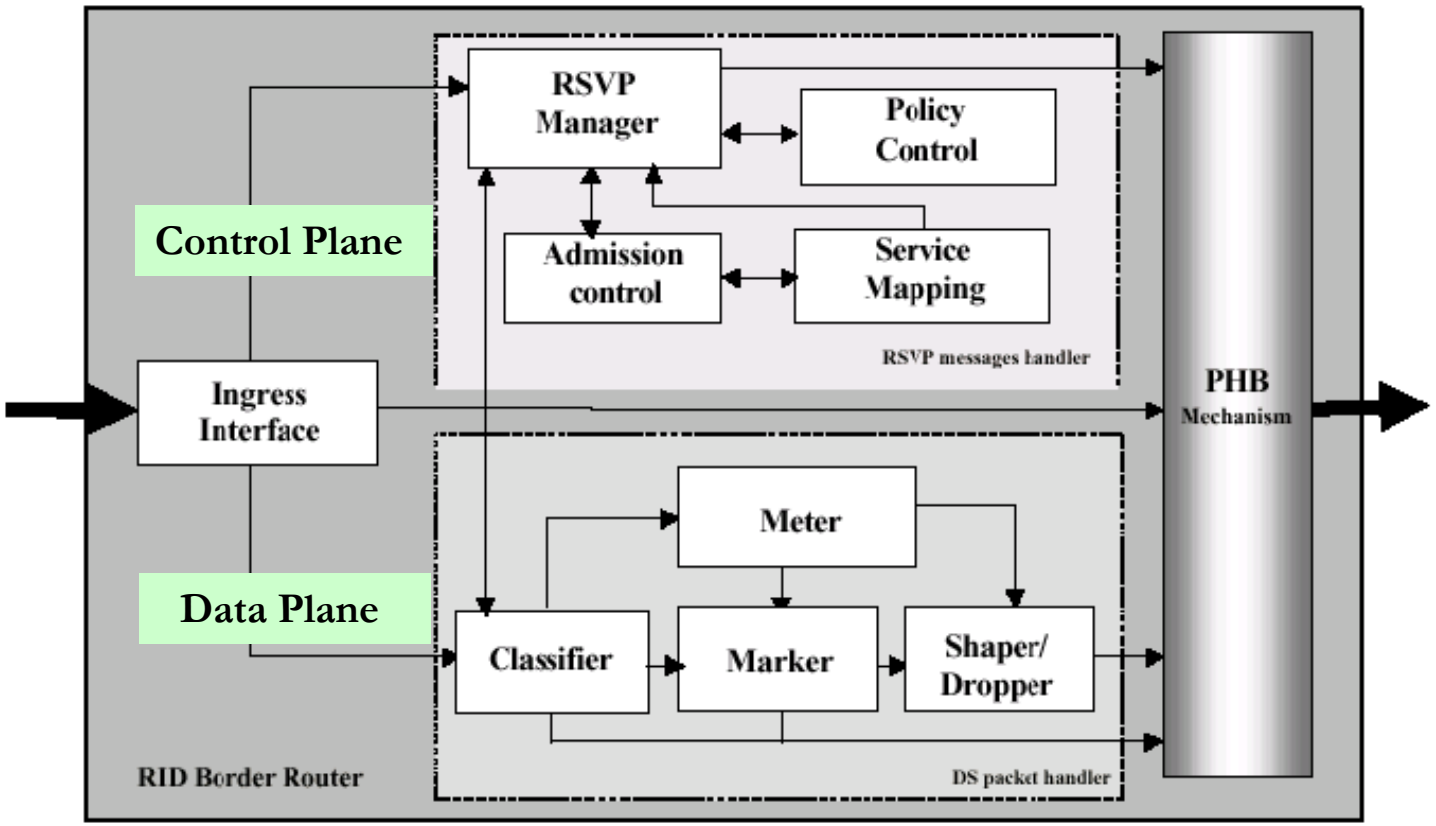
Dynamic provisioning/RSVP model

How do RSVP messages cut through the DiffServ region?

- RSVP Use: Per-flow RSVP/Aggregated RSVP/Tunneled RSVP
- Granularity of deployment of RSVP-aware routers

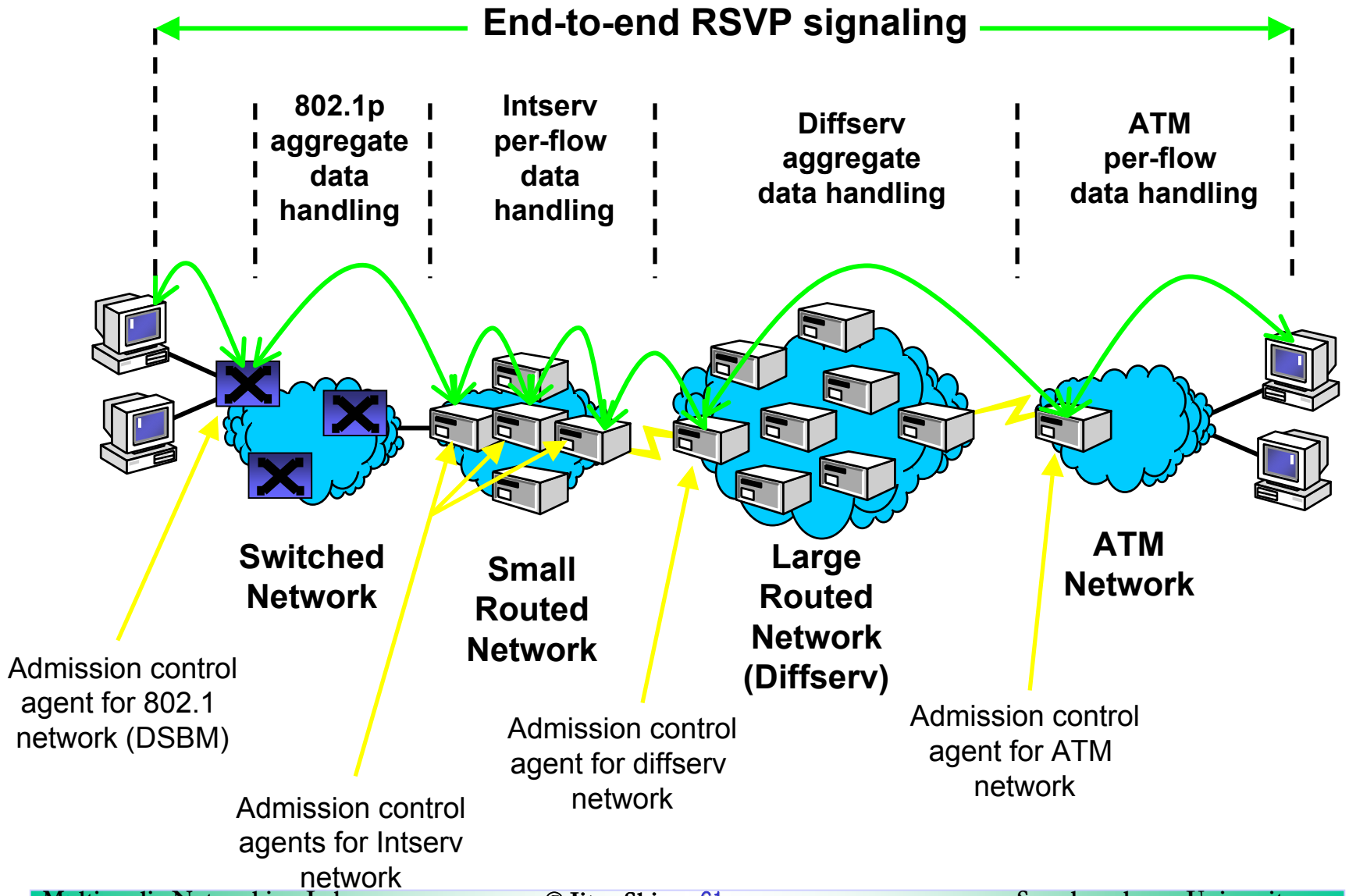


Integrated Model (4)



RSVP-aware Router Structure

Integrated Model (5)

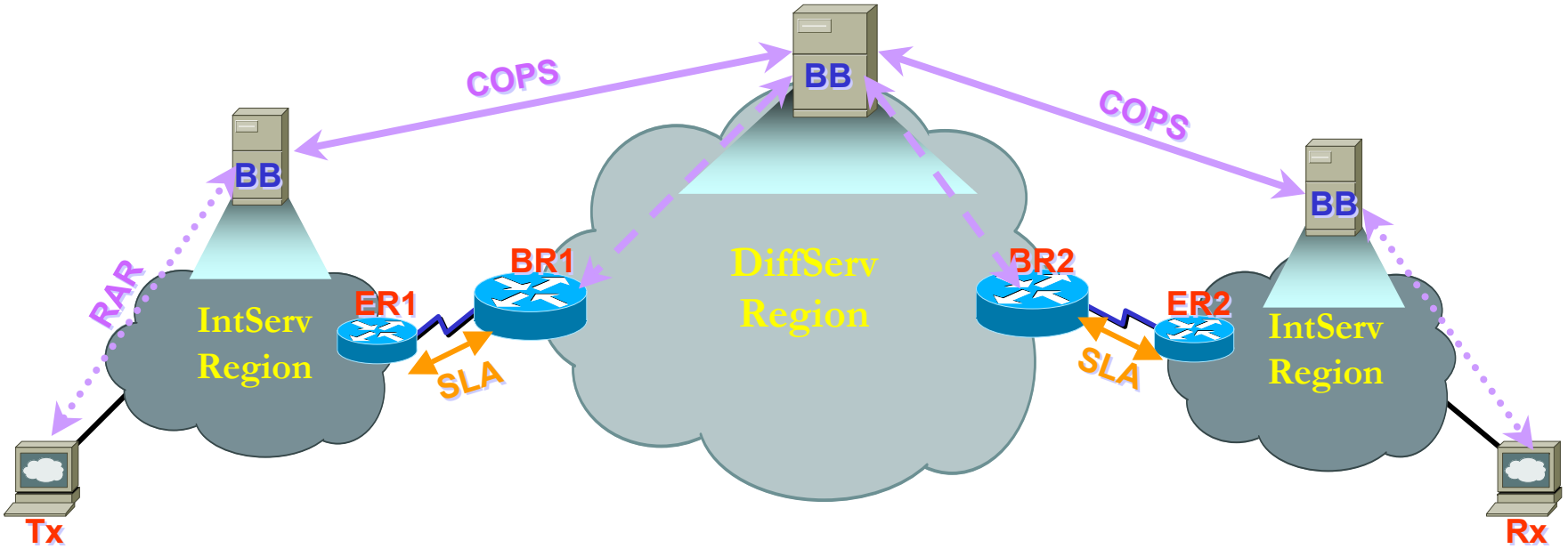


Integrated Model (6)

Dynamic provisioning/BB model

Bandwidth Broker is an centralized agent that has sufficient knowledge of resource availability and network topology to make admission control decisions.

- Non-RSVP-aware DiffServ region
- How does a BB allocate resources for a PHB?



Service Mapping (1)

How DiffServ region supports specific IntServ services

- Mapping IntServ-defined services to DiffServ-defined services(PHBs)
- Performing appropriate traffic conditioning at border routers
- Exporting IntServ parameters from the DiffServ region
- Use new RSVP DCLASS object to carry DSCP information

Service Mapping (2)

- **Proposed default mapping :** draft-ietf-issll-ds-map-01.txt
- **Another possible mapping**
  - **Network indicates alternate mapping to hosts**
    - Use RSVP signaling extensions
    - Send a signaling to sending host (If no signaling, use default)
    - Specifies DSCP which should be used to extend IntServ service type requests

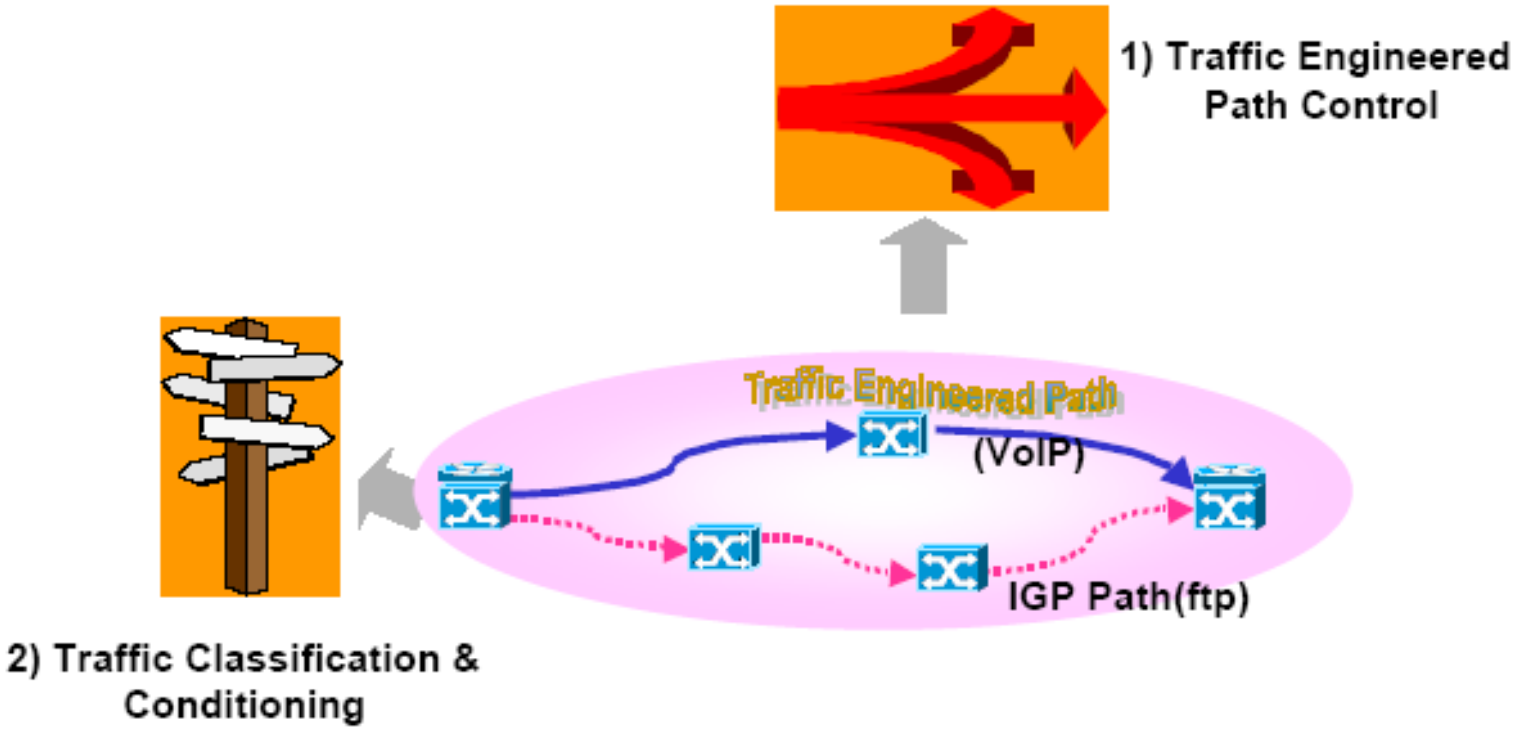
Default Mapping

IntServ Services	Delay Tolerance	DiffServ PHBs
Guaranteed Service	-	EF
Controlled Load Service	Low	AF (higher priority)
	High	AF (lower priority)

DS-aware MPLS TE : **MPLS**

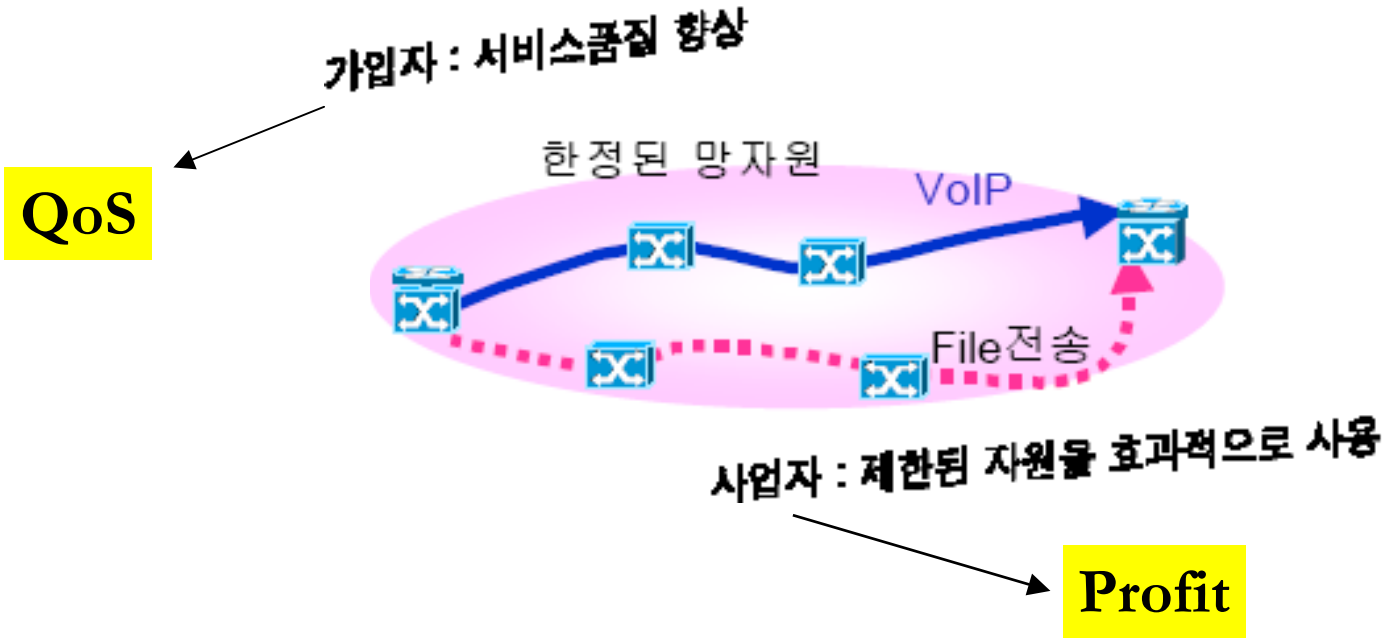
❑ MPLS Traffic Engineering

- o Traffic Engineered Path Control + Traffic Classification

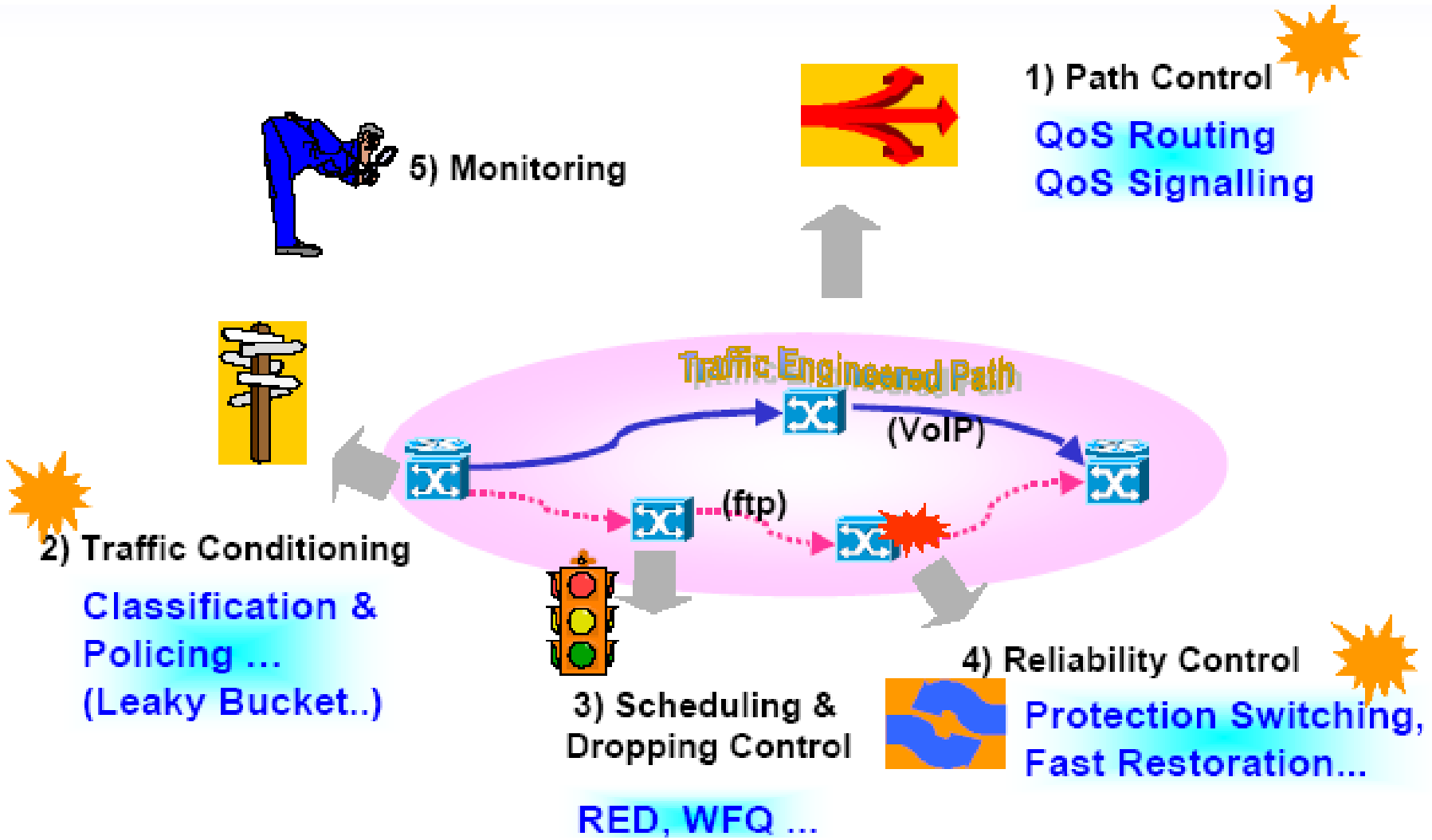


TE?

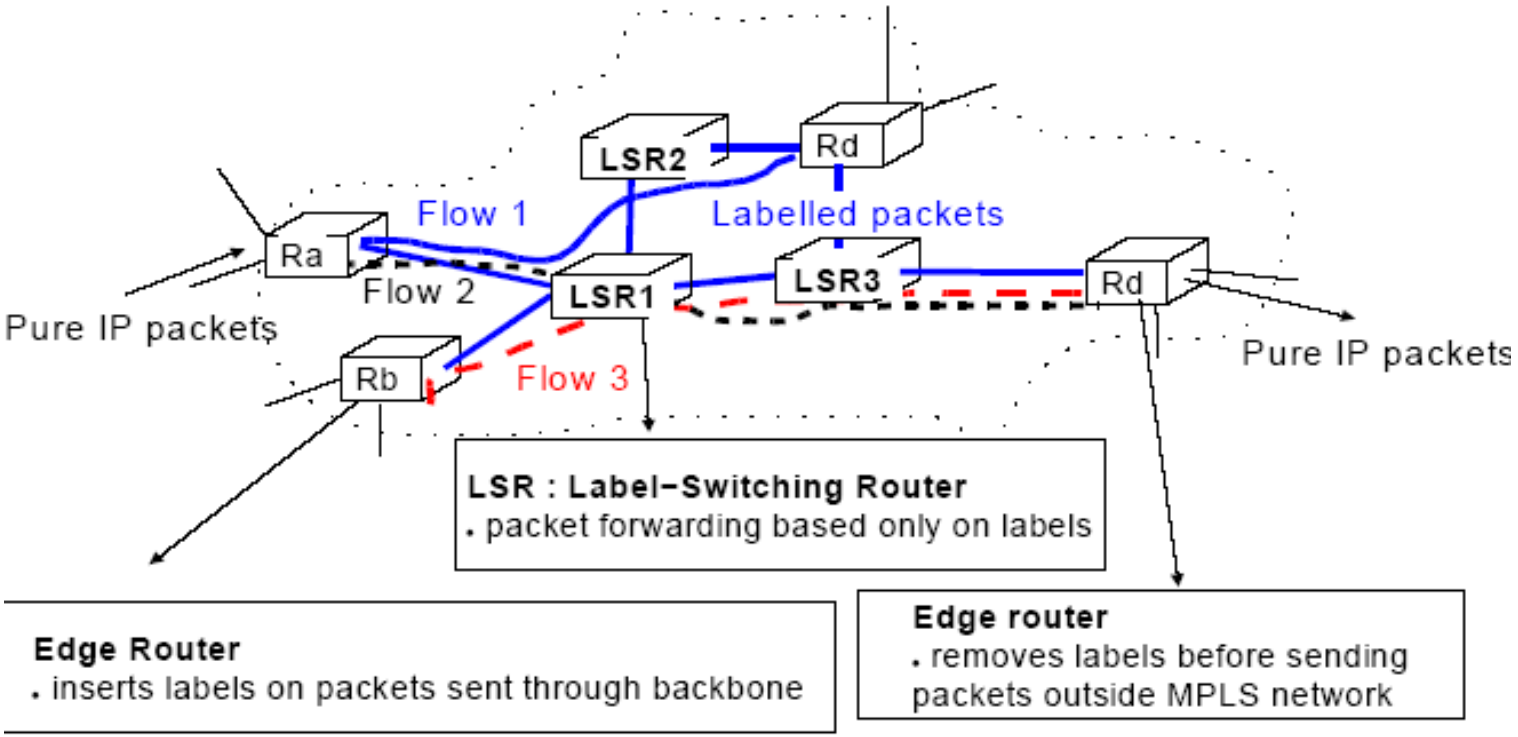
- Process of mapping traffic demand onto a network
  - 네트워크 자원 상태와 트래픽 특성, 망사업자 정책을 반영하여 가장 효과적으로 트래픽을 배치



Traffic Engineering 요소 기술



Integrating label swapping and IP



The advantage of MPLS-TE

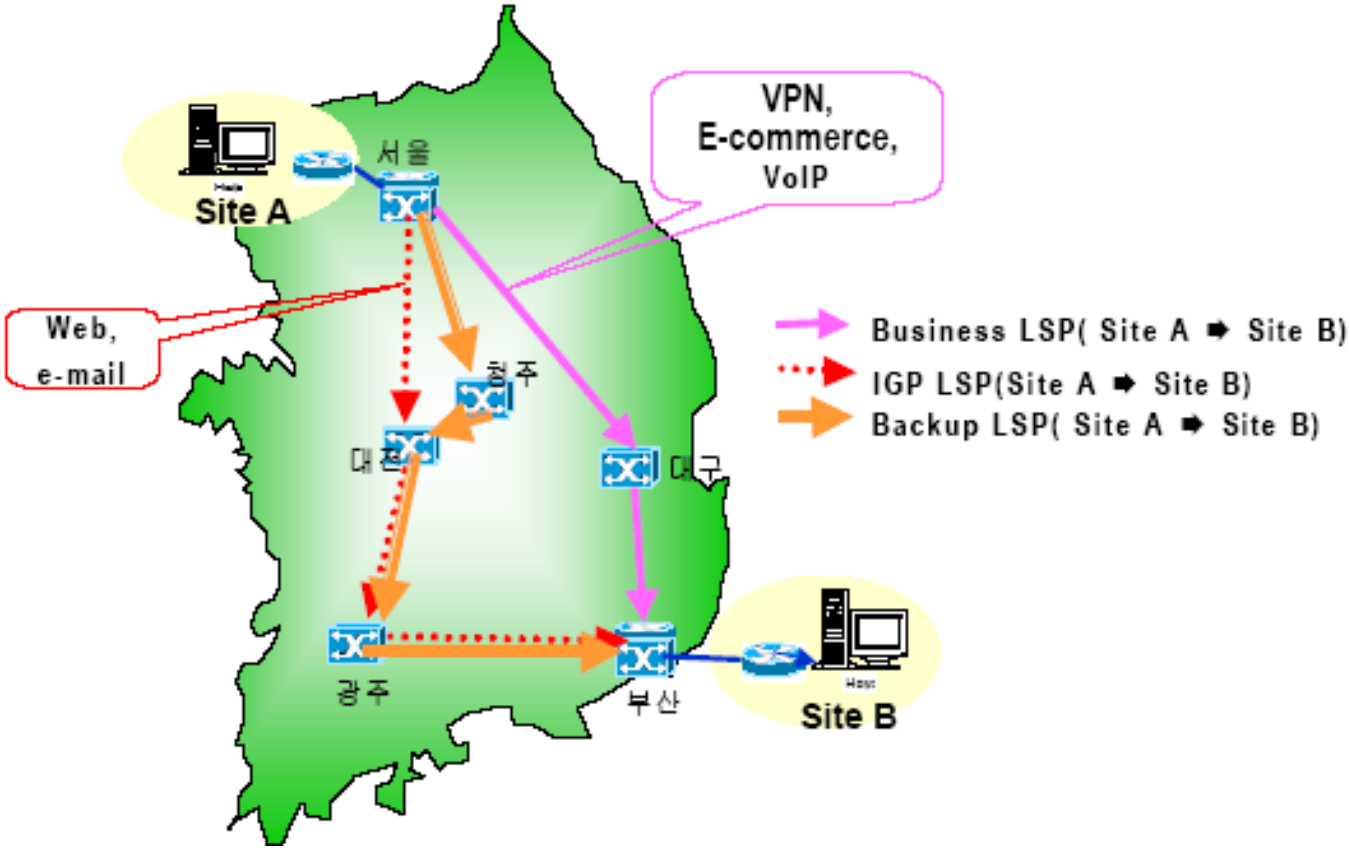
- the physical path of the "traffic-engineered path" is not limited to what the IGP would choose as the shortest path to reach the destination
- variously divisible traffic aggregation and disaggregation
- maneuvering load distribution
- stand-by secondary paths and precomputed detouring paths
- strongly unified measurement and control for each "traffic-engineered path"

QoS Path 설정 (Signalling)

- How is an TE-LSP Established ? How to coordinate the label forwarding tables of all LSRs in a given network ?
  - 시그널링 프로토콜을 이용하여 LSP를 설정/해제, 유지/관리
- Signalling Protocol
  - LDP (Label Distribution Protocol)
    - hop-by-hop LSP setup to support explicit routing
    - 세션 제어 프로토콜
  - CR(Constraint routing) -LDP
    - Extends LDP to Support Explicit Routes
    - 기능적으로 RSVP와 동일
    - **ATM 기반 ER-LSP 설정**
  - RSVP-TE
    - RSVP Extensions to Support LSP Tunnels
    - Explicit routes와 Label 정보 전달을 위한 확장 파라메터
    - **IP 기반 ER-LSP 설정**

MPLS 이용한 QoS서비스 예

- 네트워크 TE 기능을 기반으로 가입자에게 고품질 서비스 를 제공 : 특정 가입자 간에 QoS 보장형 Tunnel 서비스 제공



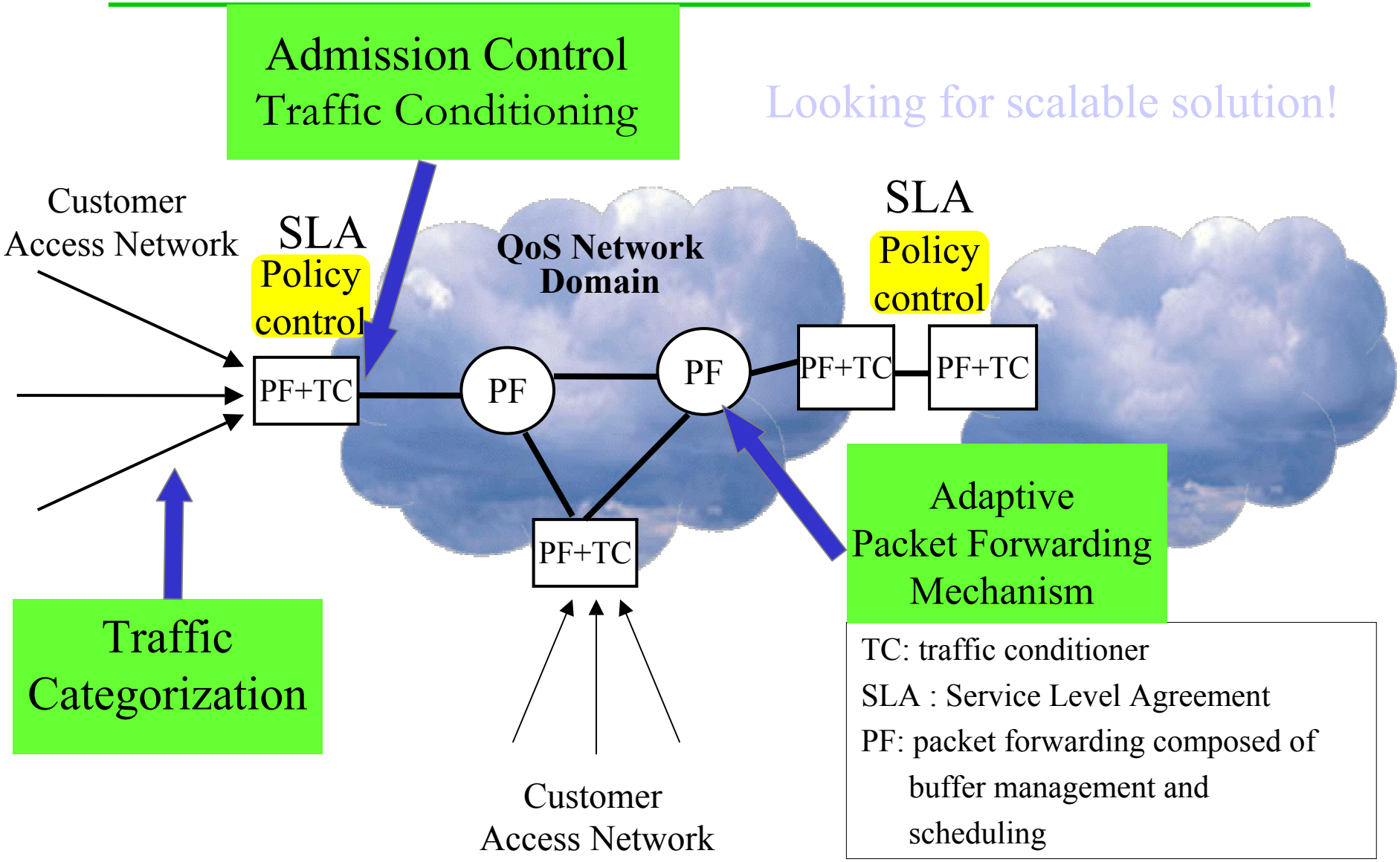
DS-aware MPLS TE

- The Trouble with MPLS TE: Admission Control
- MPLS TE is NOT QoS technology
  - Reserved bw is not guaranteed
  - RSVP-TE is solely a signaling protocol
  - does not provide traffic regulation functions of legacy RSVP
- MPLS TE designed as a tool to improve backbone efficiency independently of QoS:
  - MPLS TE compute routes for aggregates across all PHBs
  - MPLS TE performs admission control over “global” bandwidth pool for all COS/PHBs
    - i.e., unaware of bandwidth allocated to each queue

QoS Network Function Elements

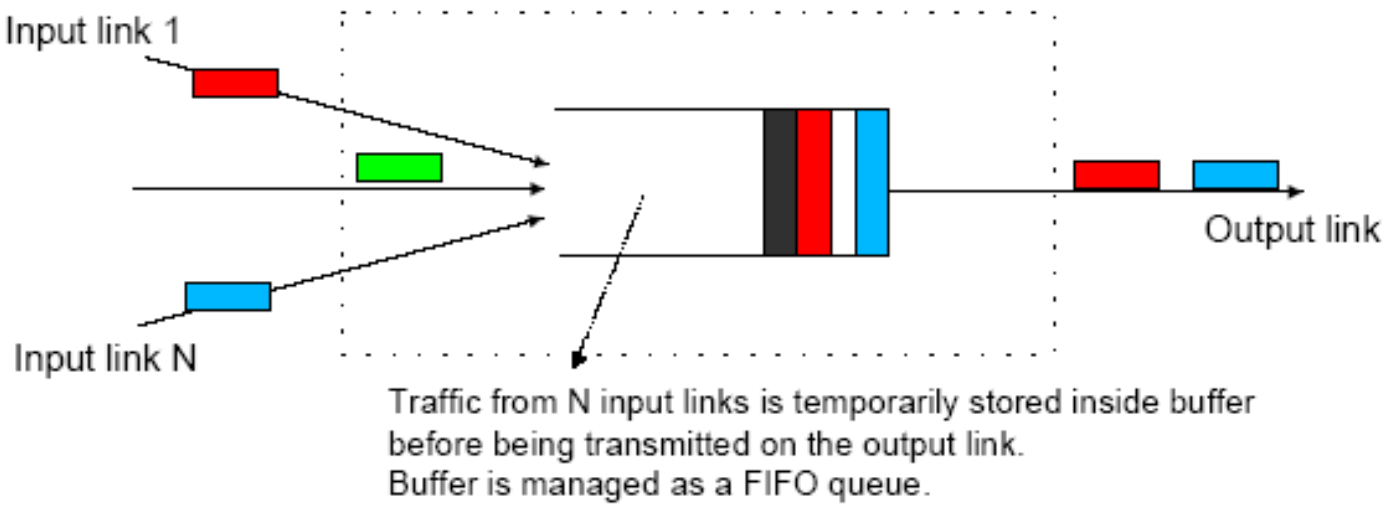
- Data plane function of network node
  - Classifier
  - Buffer management
    - Droptail, RED, etc
  - Scheduler
  - Policer/Shaper

QoS Network


Admission Control
Traffic Conditioning
Looking for scalable solution!
Customer
Access Network
SLA
Policy
control
QoS Network
Domain
SLA
Policy
control
PF+TC
PF
PF
PF+TC
PF+TC
PF+TC
Adaptive
Packet Forwarding
Mechanism
Traffic
Categorization
Customer
Access Network
TC: traffic conditioner
SLA : Service Level Agreement
PF: packet forwarding composed of
buffer management and
scheduling

Simple router (best effort service)

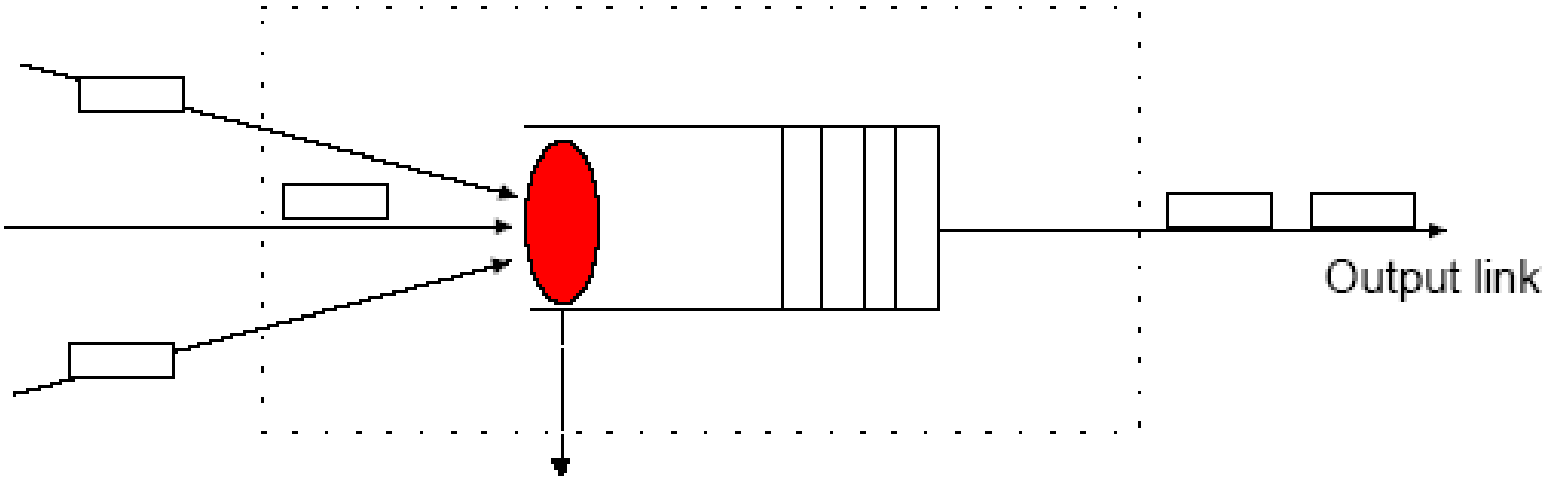
- Simplest possible router
  - N input links
  - 1 output link



- Simplest solution for best effort service

Simple router v1

- Packet treatment inside router



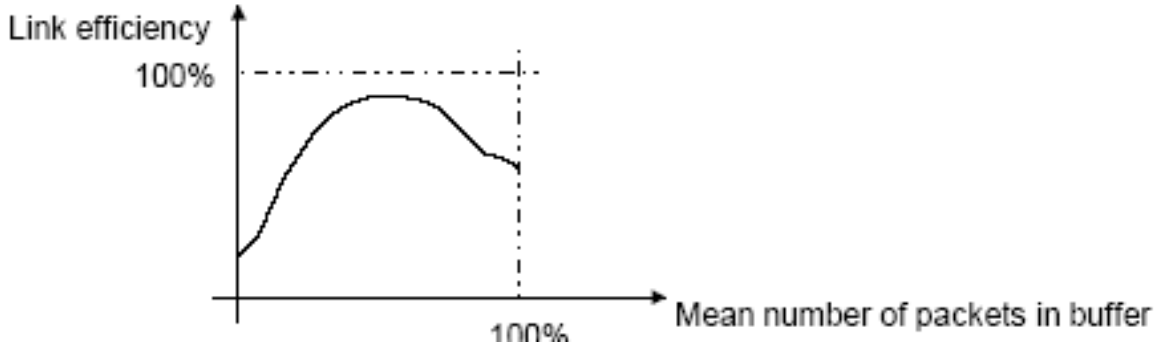
Buffer acceptance algorithm
When a packet arrives from an input link, the buffer acceptance algorithm decides whether this packet can be accepted inside the router's buffer

Buffer acceptance algorithms (1)

- Two fundamental questions
  - When do we drop a packet ?
    - when the buffer is full : (example : tail drop)
    - when the buffer occupancy increases too much (example : Random Early Detection)
- Which packet should be dropped
  - The arriving packet (the packet at the tail of the queue)
    - but is this packet responsible for congestion ?
  - Another packet from the same flow as the arriving packet
    - this might help congestion control algorithms
  - A packet from some flow
    - not necessarily from the same flow as the arriving packet

Buffer acceptance algorithms(2)

- Objectives
  - control the amount of packets in the buffer to
    - efficiently support best−effort traffic, should provide a fair utilization of the routers buffers
    - provide protection among different flows, one flow should not prohibit other flows from having packets inside the router's buffers
    - achieve a good utilization of output link



Drop Tail

- Simplest buffer acceptance algorithms
- Principle
  - when a packet arrives at a full buffer, the arriving packet is discarded
  - Advantages
    - easy to implement
    - can limit the number of packet losses for large buffer
  - Disadvantages
    - no distinction between the various flows
    - not the best solution for TCP traffic

Random Early Detection (RED) : **Goals**

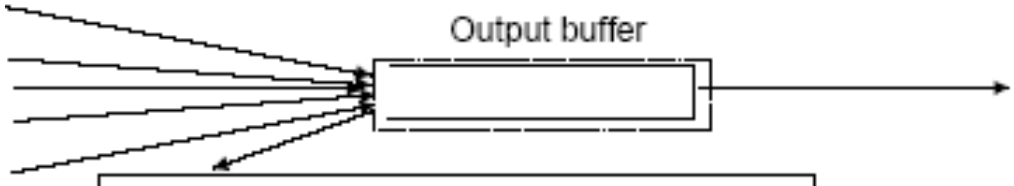
- should be easily implemented in simple routers with a single logical queue
- achieve a low, but non−zero, average buffer occupancy
  - low average occupancy provides low delay for interactive applications and ensure fast TCP response
  - non−zero average occupancy ensures an efficient utilization of the output link
- approximate a fair discard of packets among the active flows without identifying them
- discard packets in a TCP friendly way
  - we should avoid discarding bursts of packets since TCP reacts severely to burst losses

RED :Principle

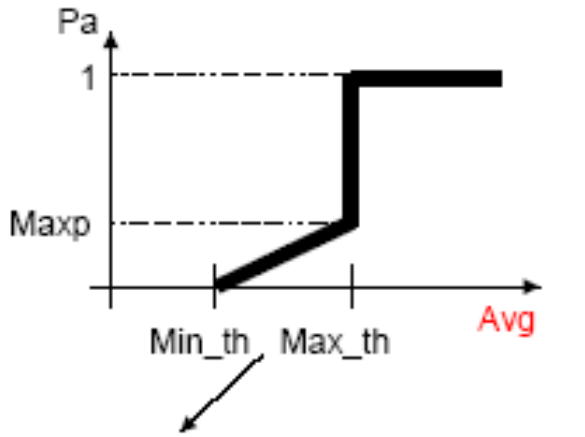
- How can we detect congestion ?
  - measure average buffer occupancy by using a low－pass filter
  - buffer is considered congested when its average occupancy is above a configured threshold : threshold value usually around 10%－ 20% of buffer size
- What do we do in case of congestion ?
  - Probabilistic drop for incoming packet
    - drop will force TCP to slow down
    - drop probability should increase with congestion level
- Why probabilistic drop ?
  - Avoid dropping burst of packets from single flow
  - Try to drop packets for each flow in proportion of network usage
  - Avoid synchronization effects

RED

- Implementation
  - suitable for routers with a single queue




```
Packet arrival :
compute average queue occupancy, avg
if (avg < min_th)
        // no congestion
        accept packet
else if (min_th ≤ avg < max_th )
        // near congestion, probabilistic drop
        calculate probability Pa
        with probability Pa
                discard packet
        else with probability (1−Pa)
                accept packet
        else if avg ≥ max_th
                discard packet
```



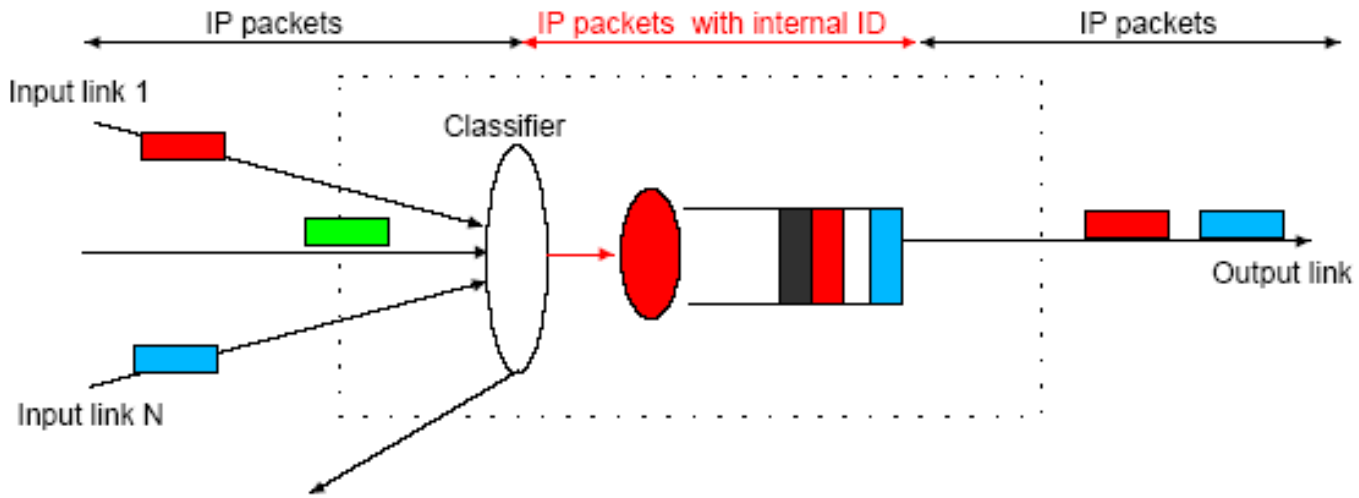
Issues with RED

- Difficult to provide a clear answer today
  - Some argue that RED provides
    - a better network utilization
    - a lower queuing delay
- Others complain on the complexity of tuning RED
  - How do we set min_{th}, max_{th}, max_p and w_q in an operational network ?
    - Do the settings depend on link speed, type of traffic, ... ?
  - A bad choice of the RED parameters may provide a worse performance than plain old tail−drop

Simple router v2 : classification

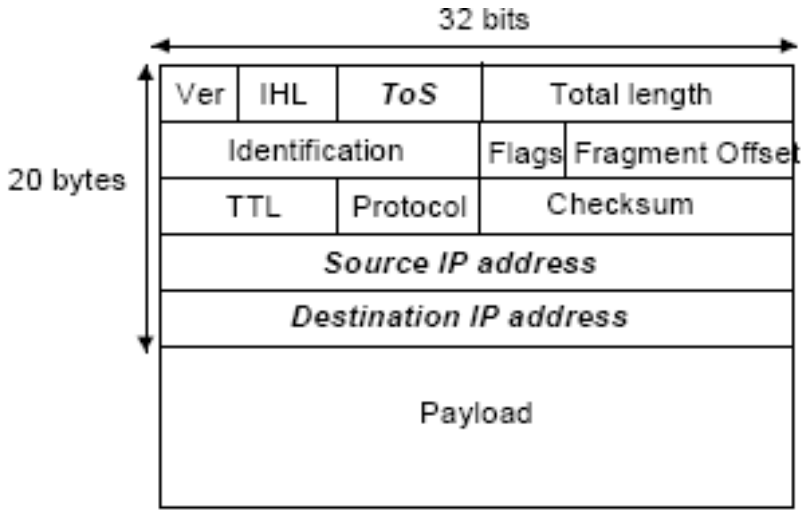
- Roles of the classifier
  - identify the flow to which an arriving packet belongs
    - identification can require complex operations
  - store this information internally so that other parts of the router will easily determine the flow of a packet
    - classification should be done at most once in each router

Flow?



Flow identifier

- Layer – N flow?
  - layer two flow
    - e.g. ATM or frame relay circuits
  - layer three flow [IP related]
  - layer four flow [TCP or UDP related]
  - layer seven flow [application level flow]



- Identification of layer−three flows
  - source and destination IP addresses with or without associated netmasks : e.g. all traffic from 138.48.0.0/16
  - all IP traffic with same route or BGP next hop
    - requires a route table lookup by the classifier

Classification Issues

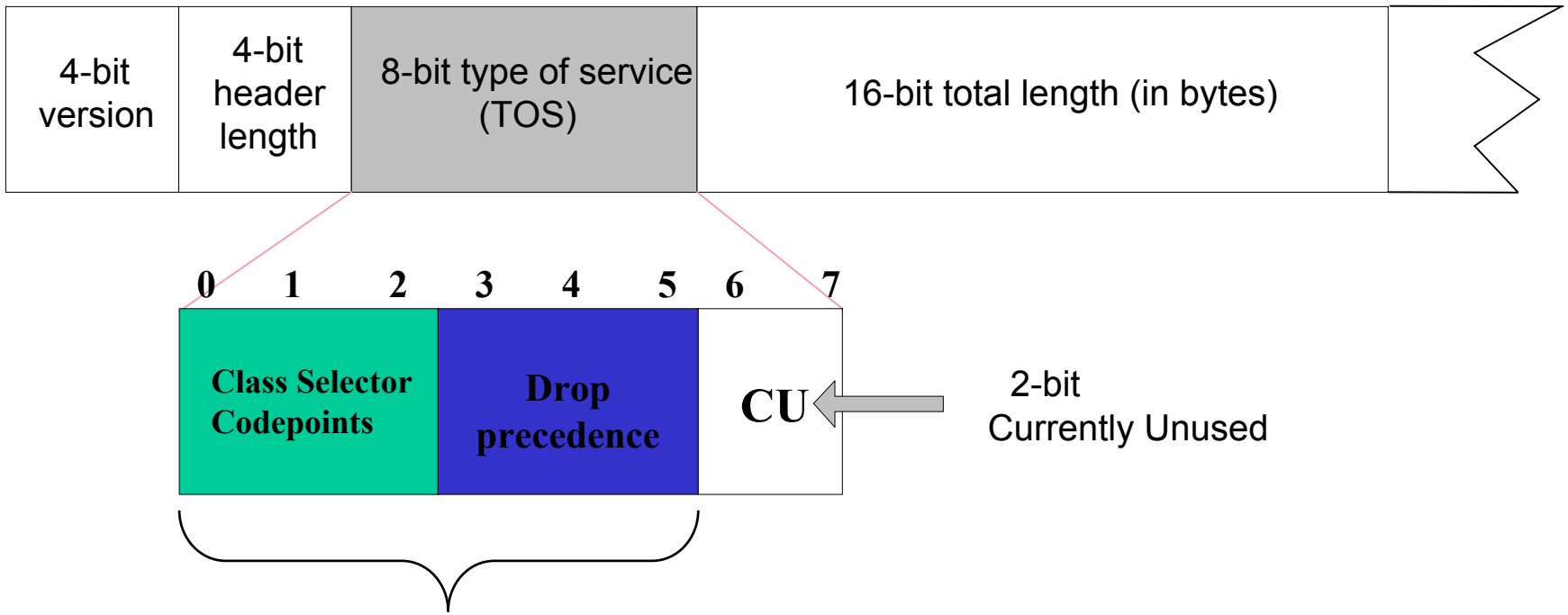
- At which layer should we classify ?
- layer−7 classification is very expensive
  - requires examination of packet headers and contents
  - will probably only be used by special equipments
    - will not work at high−speed
    - will not be deployed in backbones
- layer−3 versus layer−4 classification
- no real consensus today
- some believe that layer−4 classification can be performed by each router, even in backbones
  - ASICs required to perform layer−4 classification at high speed
- many others believe that backbone routers cannot perform complex layer−3 and layer−4 classifications
  - classification should be done by edge routers
  - backbone routers should only look at special markings

IP packet marking (1)

- How can we mark an IP packet ?
  - Steal one field of the IP header
    - ToS : Type of Service Octet
    - defines the relative importance of the IP packet and the type of service required for this packet
- current status
  - definition of ToS Octet changed several times
  - Precedence is used in some networks
  - ToS field is rarely used
- Using the ToS Octet for marking
  - advantage : easy to implement
  - disadvantage : limited number of marked flows

TOS → DSCP

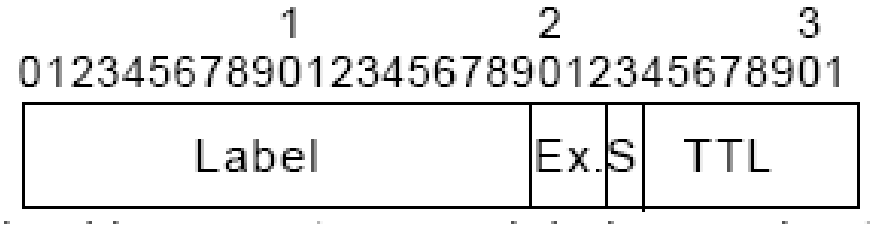
IPv4 Header (first 32 bits)



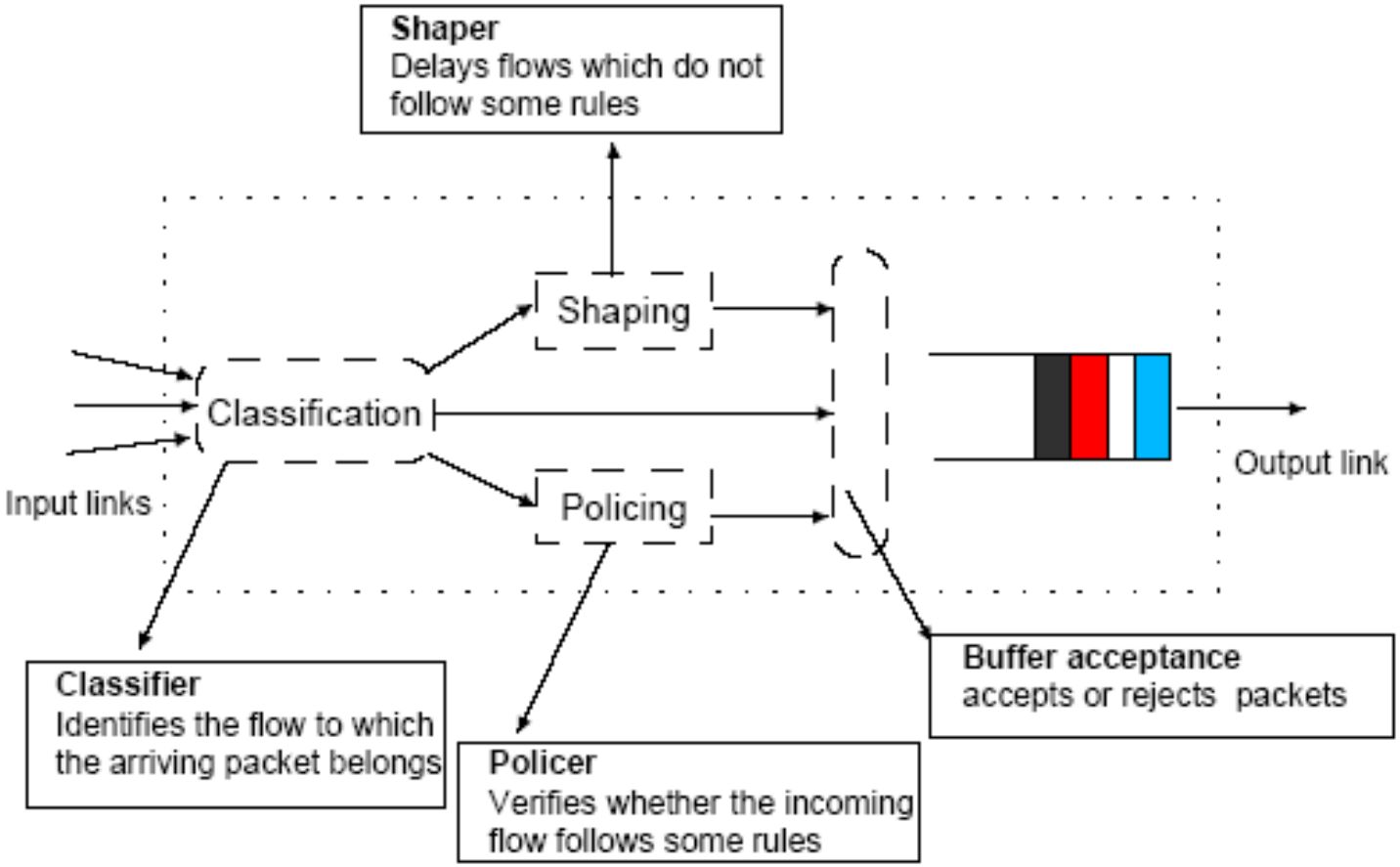
* RFC2597 : 4 classes and 3 drop precedence levels for AF PHB
* RFC2598 : a single codepoint for EF PHB

IP packet marking (2)

- How can we mark an IP packet ?
  - rely on the layer 2 protocols
  - layer−3 flows mapped on different layer 2 flows
  - QoS will be provided by layer 2
- Insert a new header in front of the IP packet
  - Multiprotocol Label Switching (MPLS)
  - Principle : edge routers identify layer−3 flows and insert one 32 bits MPLS header in front of each IP packet from each flow



Simple router v3

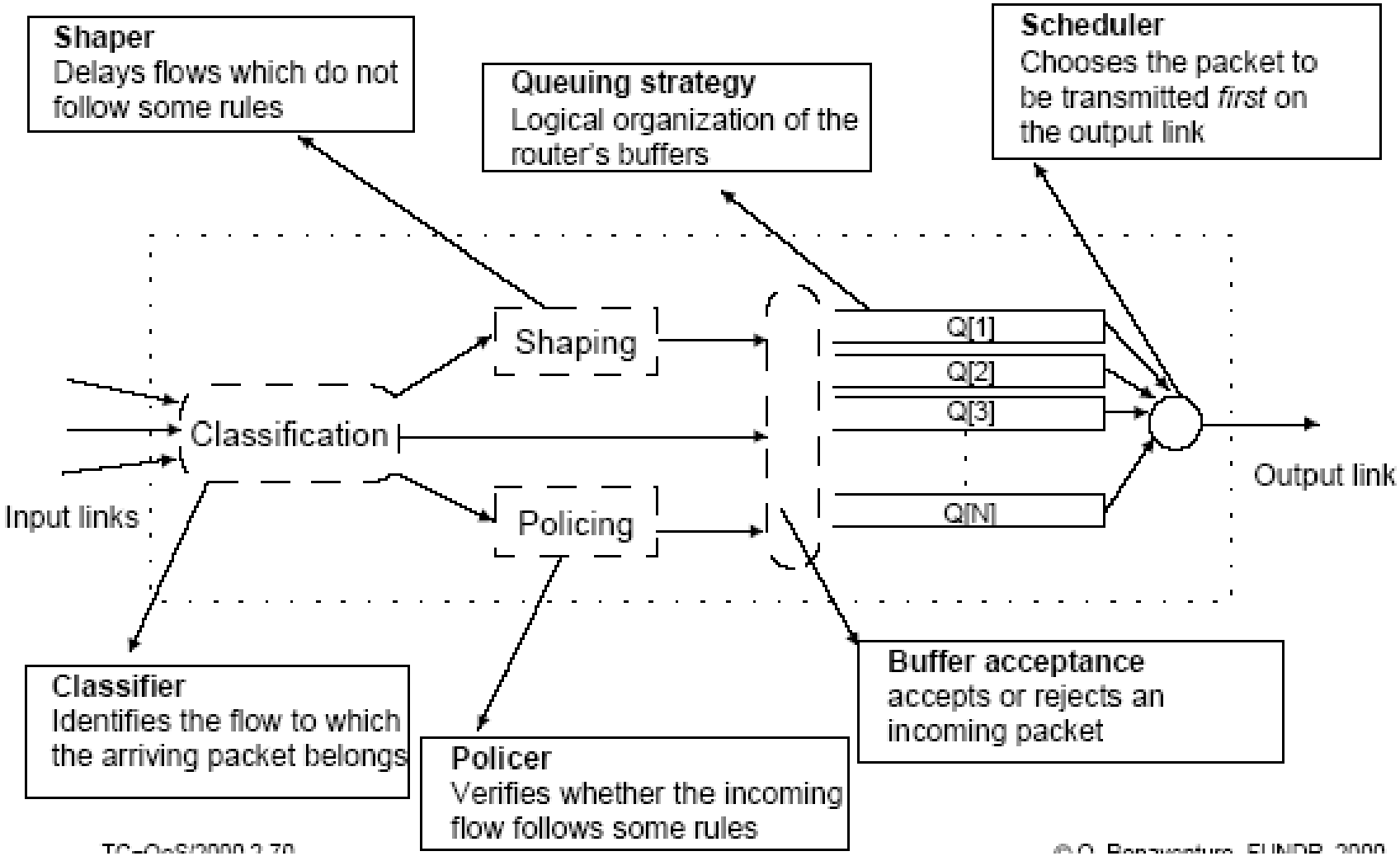


Marker
Meter
dropper

Traffic Conditioning

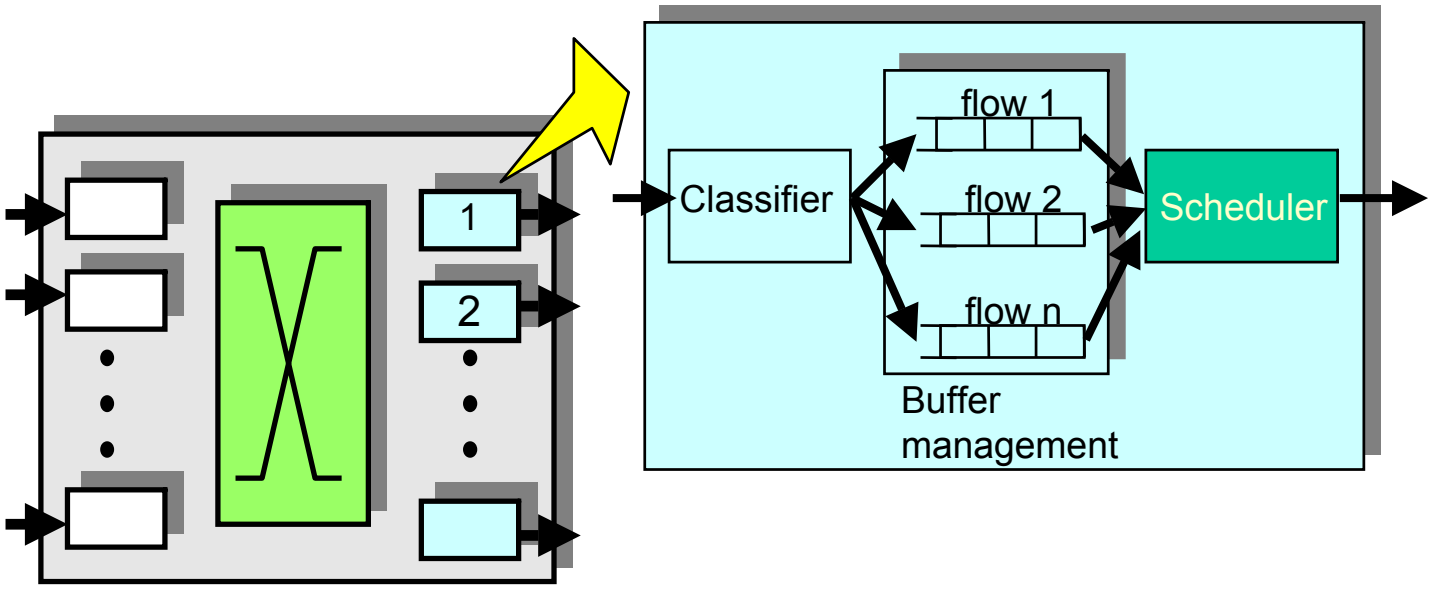
- Shaping & Policing
- Example of marking
  - Marking based on flow id and current rate
    - best effort packets are low priority
    - in excess packets from min bandwidth flows are low priority
    - non－excess packets from min bandwidth flows are high priority
    - max bandwidth packets are high priority
- Examples of metering
  - Token bucket-based
  - Time sliding window-based

Scheduler


Shaper
Delays flows which do not follow some rules
Queuing strategy
Logical organization of the router's buffers
Scheduler
Chooses the packet to be transmitted *first* on the output link
Shaping
Classification
Policing
Q[1]
Q[2]
Q[3]
Q[N]
Input links
Output link
Classifier
Identifies the flow to which the arriving packet belongs
Policer
Verifies whether the incoming flow follows some rules
Buffer acceptance
accepts or rejects an incoming packet

Packet Scheduling

- Decide when and what packet to send on output link
  - Usually implemented at output interface



Scheduler

- Function
  - among all the logical queues containing at least one packet, select the packet that will be transmitted on the output link
- A scheduler should ...
  - be easy to implement in hardware
  - support best−effort and guaranteed services
  - provide fairness for best−effort traffic
  - provide protection
    - one flow should not be able to steal bandwidth from other existing flows
  - provide statistical or deterministic guarantees
    - bandwidth, delay

Types of Scheduling algorithms

- Work−conserving scheduler
  - a work−conserving scheduler will always transmit one packet provided that there is at least one packet inside the router buffers
- Non−work−conserving scheduler
  - a non−work−conserving scheduler may defer the transmission of packets on the output link even if some packets are waiting inside the router buffers
  - can provide guarantees on delay jitter
  - nice in theory, but not often implemented

Why Packet Scheduling?

- Can provide per flow or per aggregate protection
- Can provide absolute and relative differentiation in terms of
  - Delay
  - Bandwidth
  - Loss

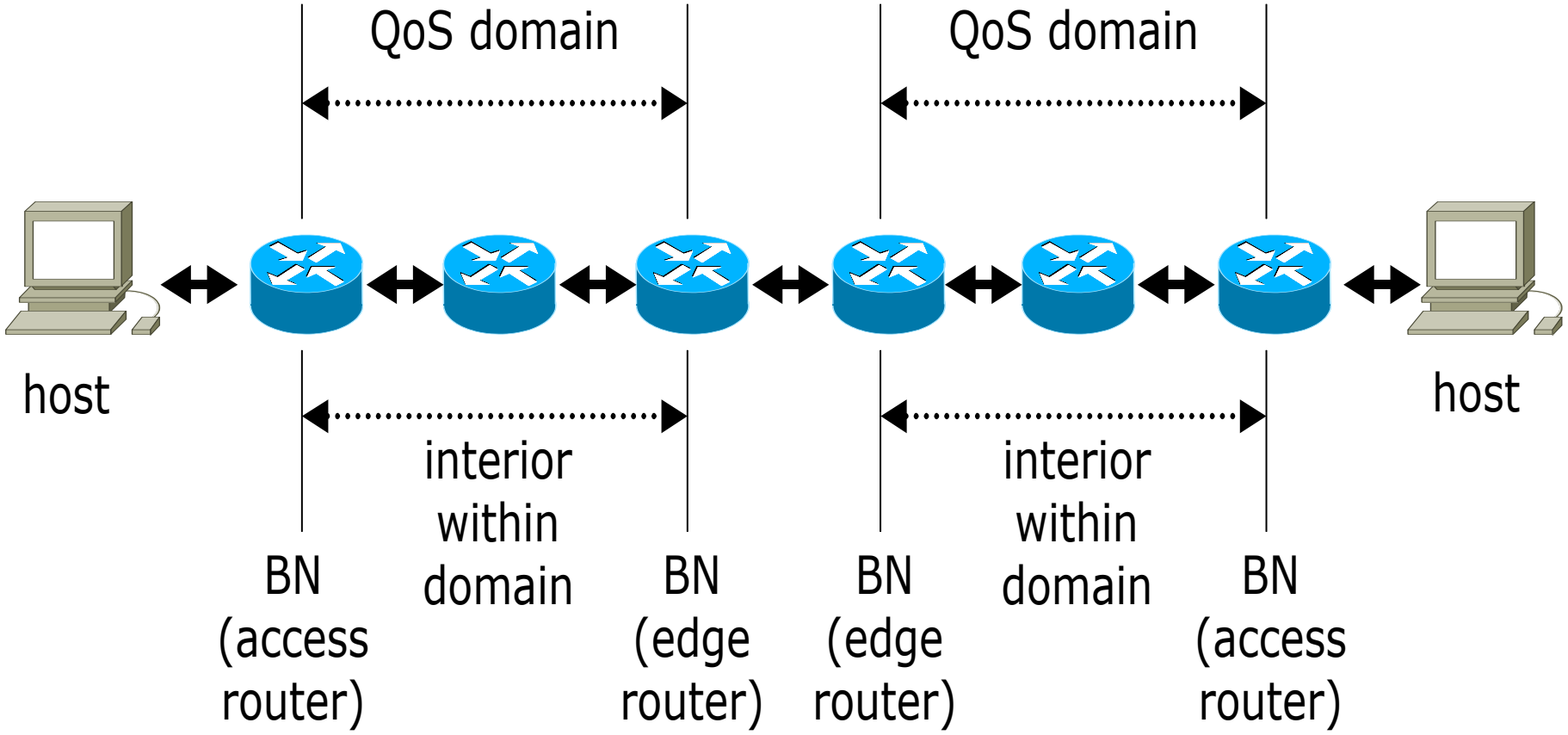
Fair Queueing

- In a fluid flow system it reduces to bit-by-bit round robin among flows
  - Each flow receives $min(r_i, f)$, where
    - r_i – flow arrival rate
    - f – link fair rate (see next slide)
- Weighted Fair Queueing (WFQ) – associate a weight with each flow
  - In a fluid flow system it reduces to bit-by-bit round robin
- WFQ in a fluid flow system → Generalized Processor Sharing (GPS)

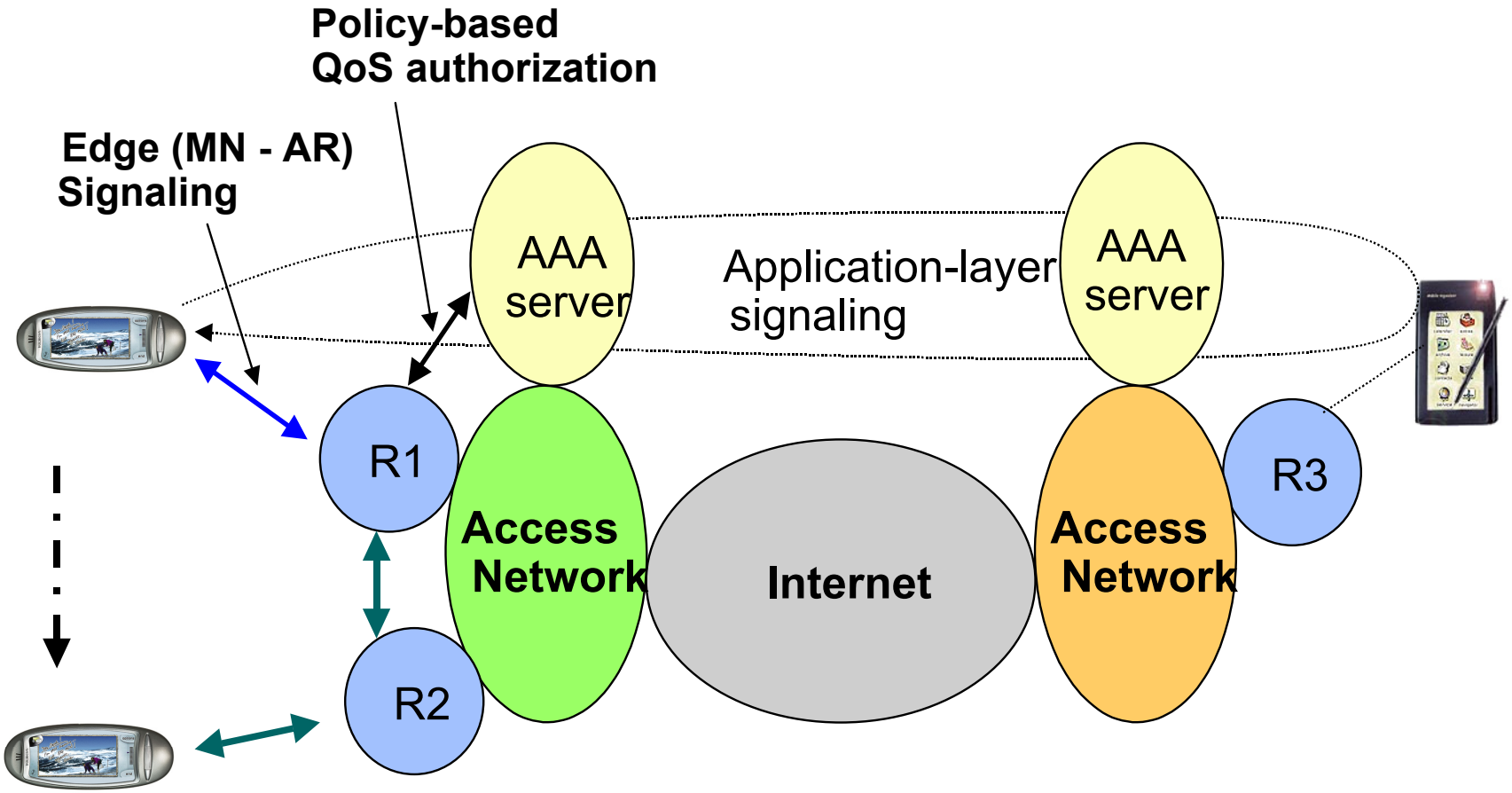
End-to-End QoS

- Why end-to-end QoS?
- What is missing to get end-to-end (E2E) QoS to work?
  - Performance problems with implementations
  - Business models lacking
  - No commonly agreed architecture, E2E QoS architecture, signaling, etc.
    - QoS signaling needed to reserve and release resources across different network environments, such as across administrative and/or technology domains

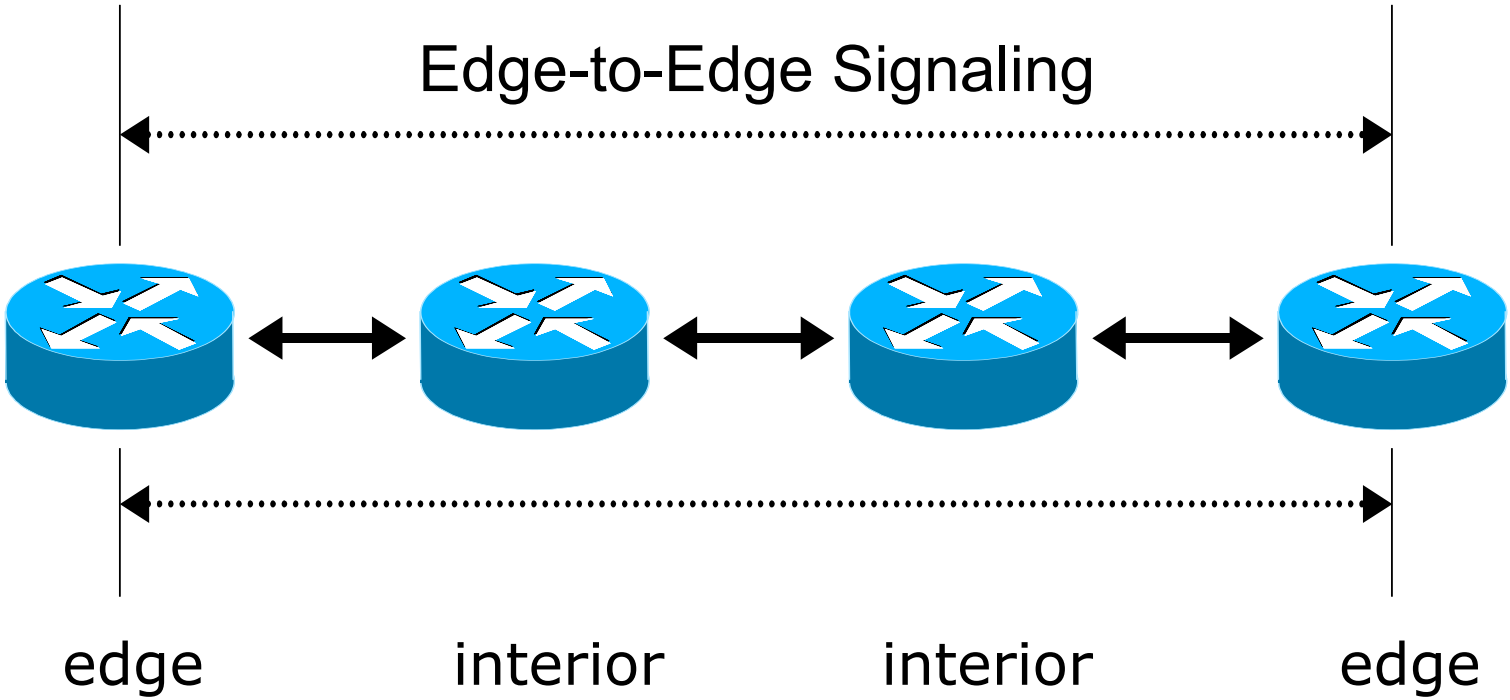
End-to-End QoS Architecture



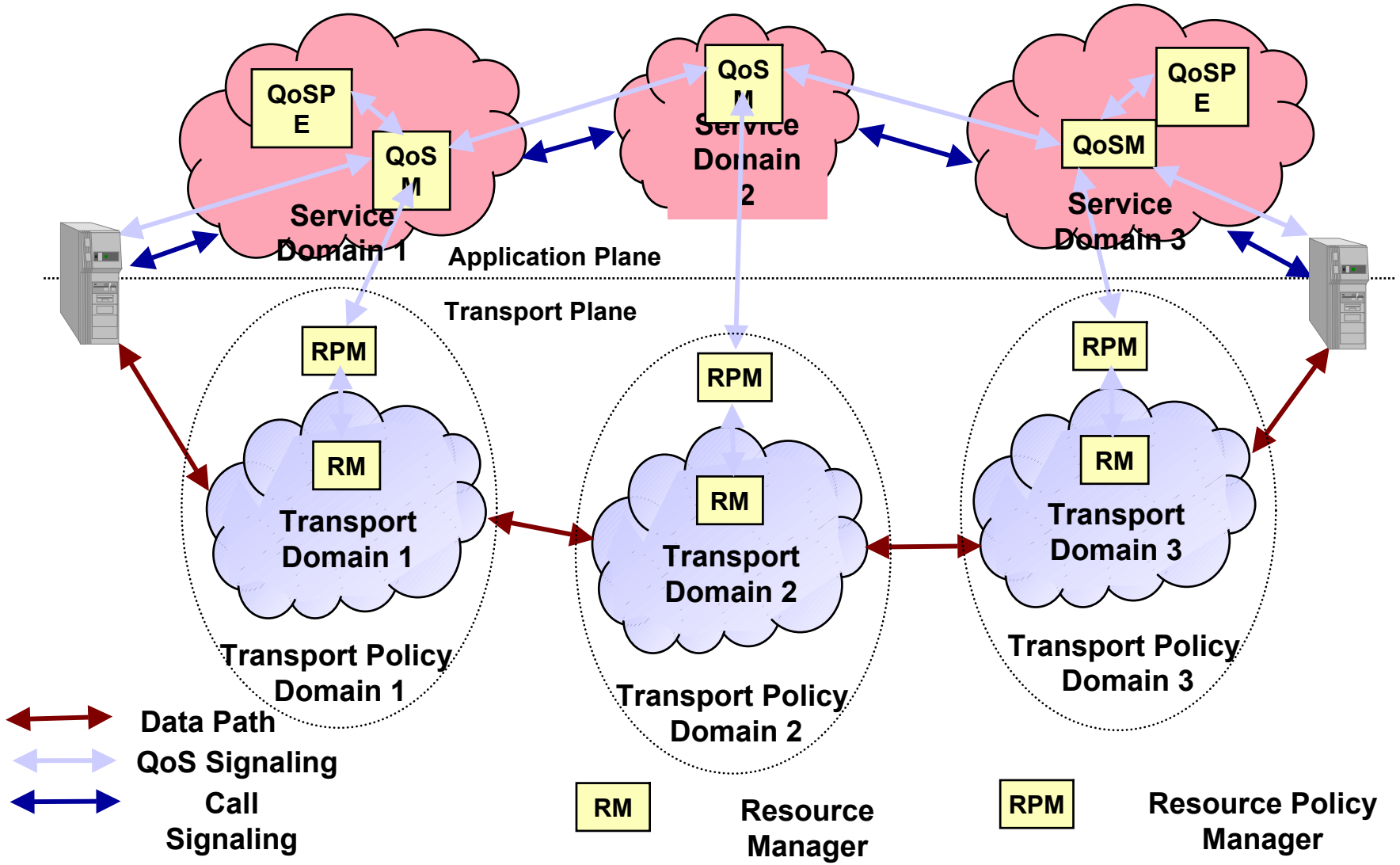
An Edge Signaling Example



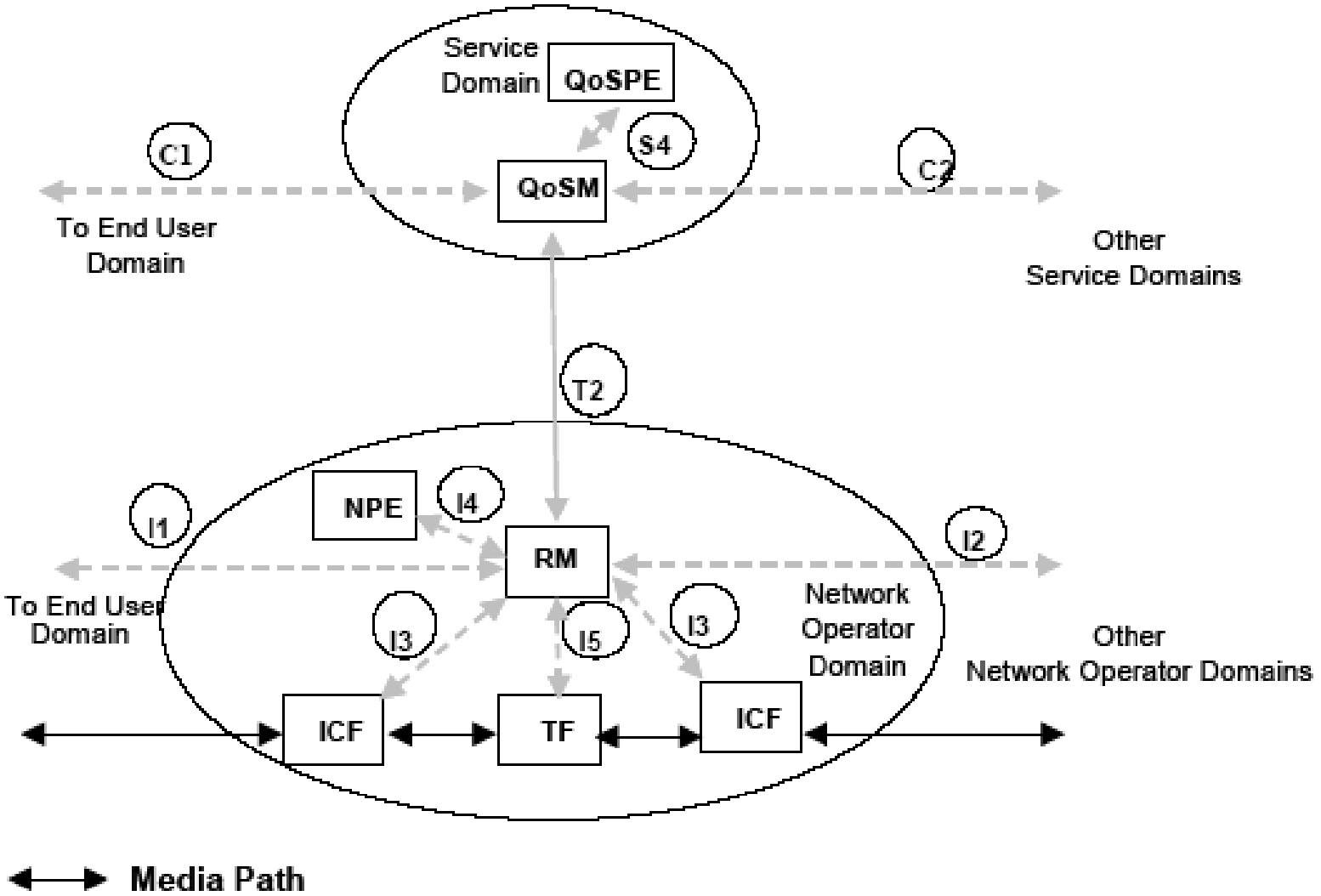
An Edge-to-Edge Signaling Example



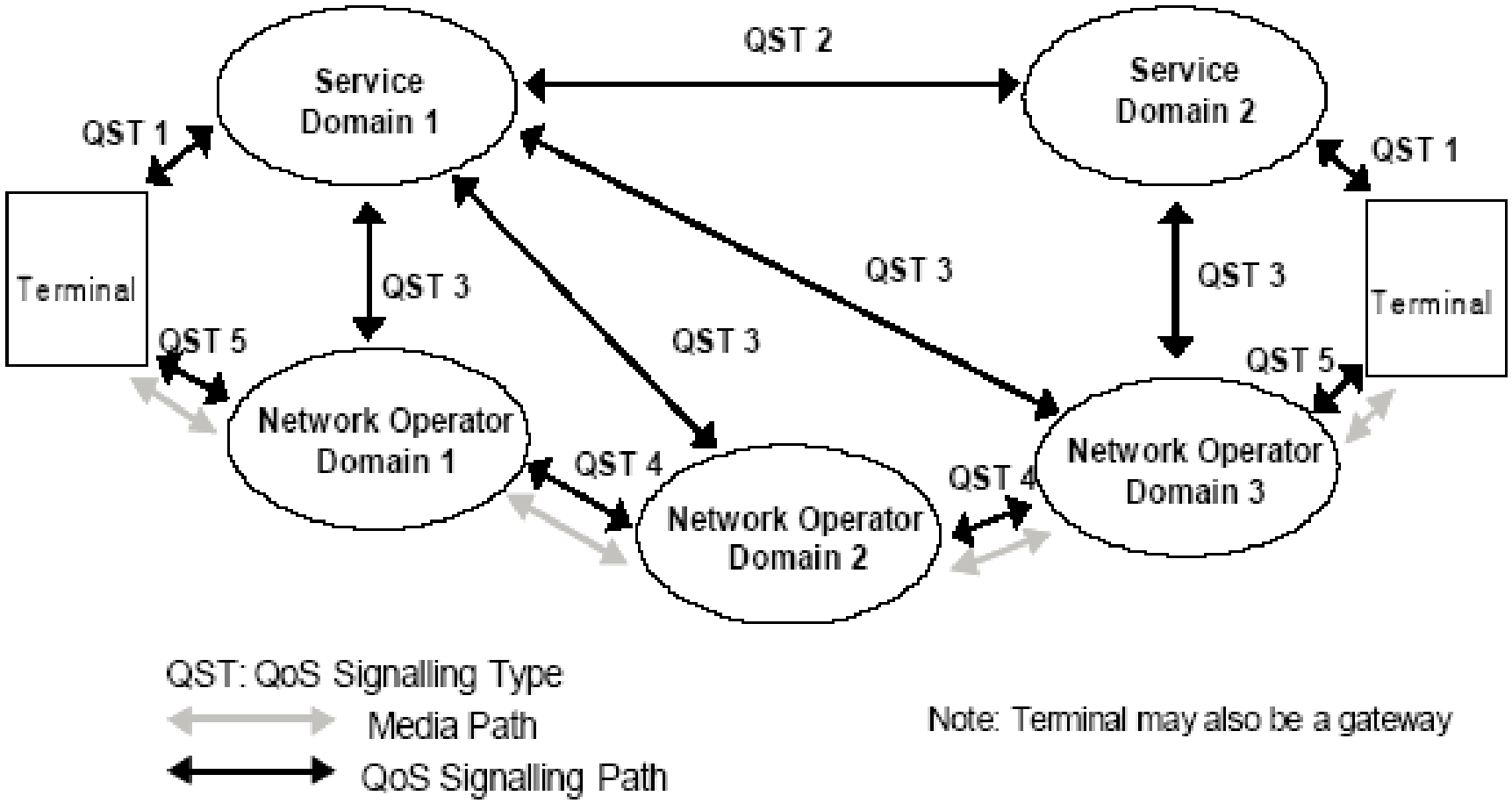
Service Domain Based QoS Signaling



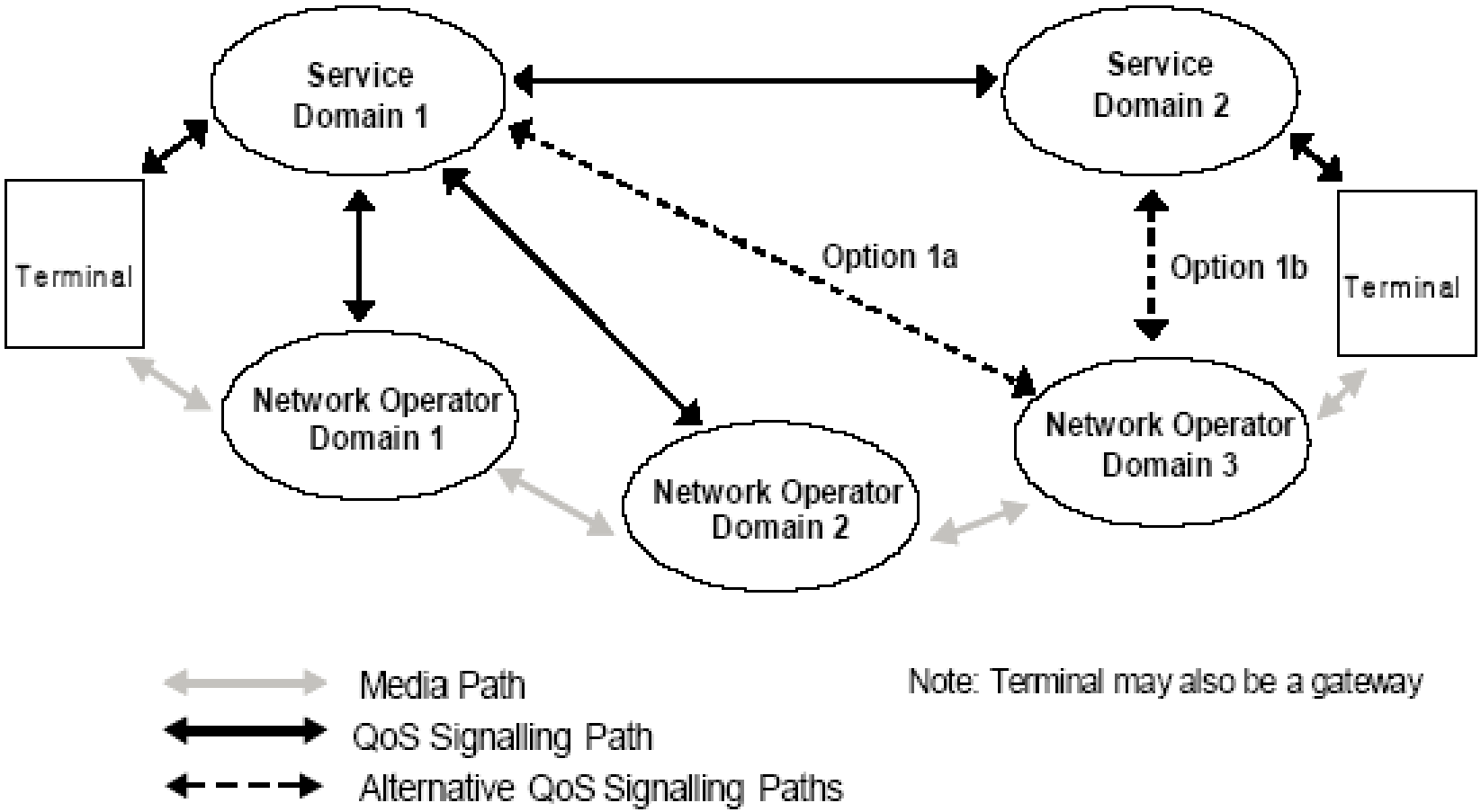
An Architecture for End-to-End QoS Control and Signaling (ITU-T SG16)



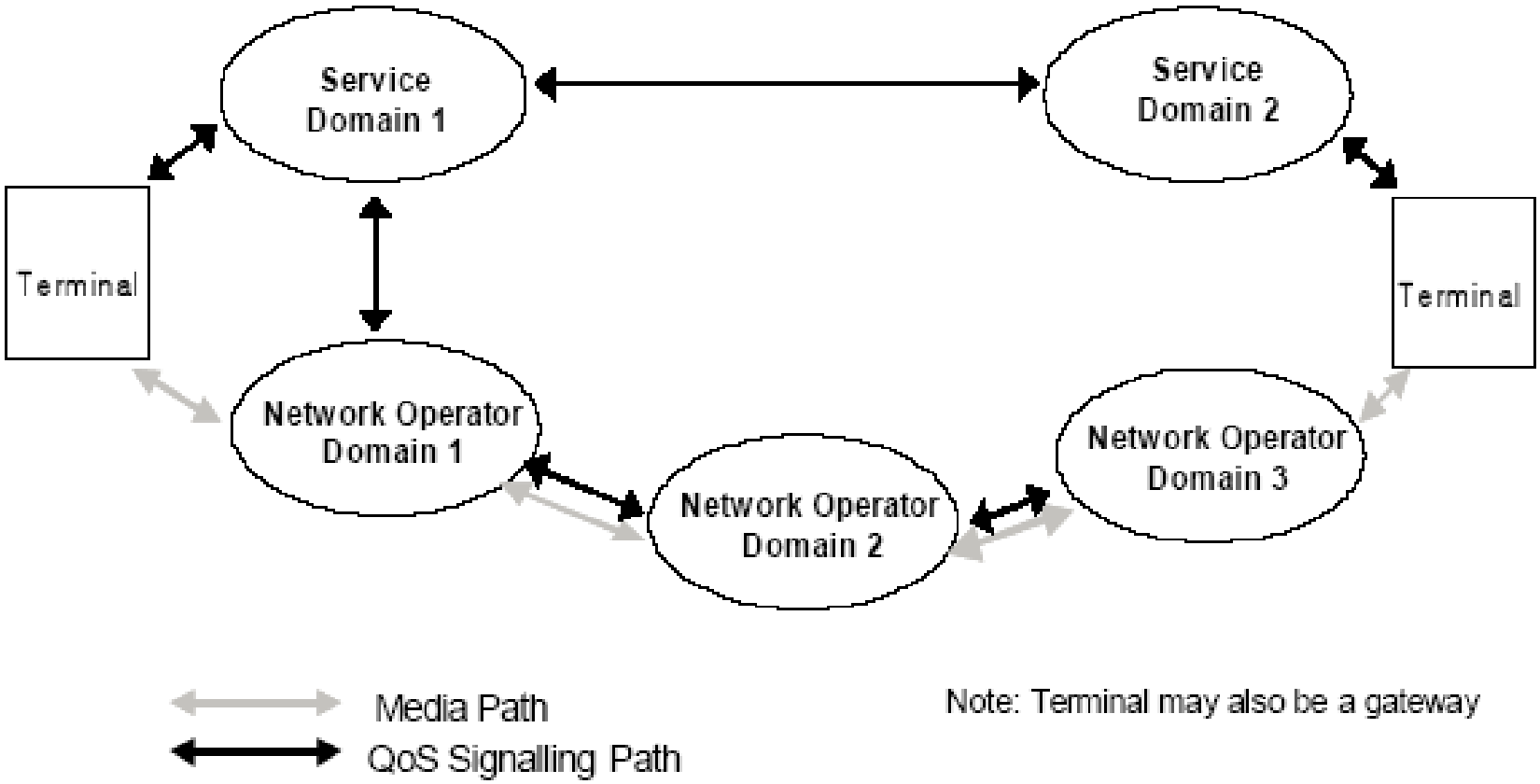
Classification of QoS Signaling Types (ITU-T)



Option 1: ASP Controlled Inter-Domain Routing (ITU-T)



Option 2: Network Operator Controlled Inter-Domain Routing (ITU-T)



Existing QoS (Signaling) Solutions

- Based on draft-demeer-nsis-analysis-03.txt
- End-to-end per-flow resource reservation protocol:
  - Resource Reservation Protocol (RSVP)
- Integrated Services over Differentiated Services:
  - Framework specified in RFC 2998
- Statically assigned trunk reservations based on Differentiated Services:
  - Framework specified in RFC 2475
- Dynamic trunk reservations with Aggregated RSVP:
  - Specified in RFC 3175
- Traffic Engineering Tunnels and RSVP
  - RSVP-TE specified in RFC 3209

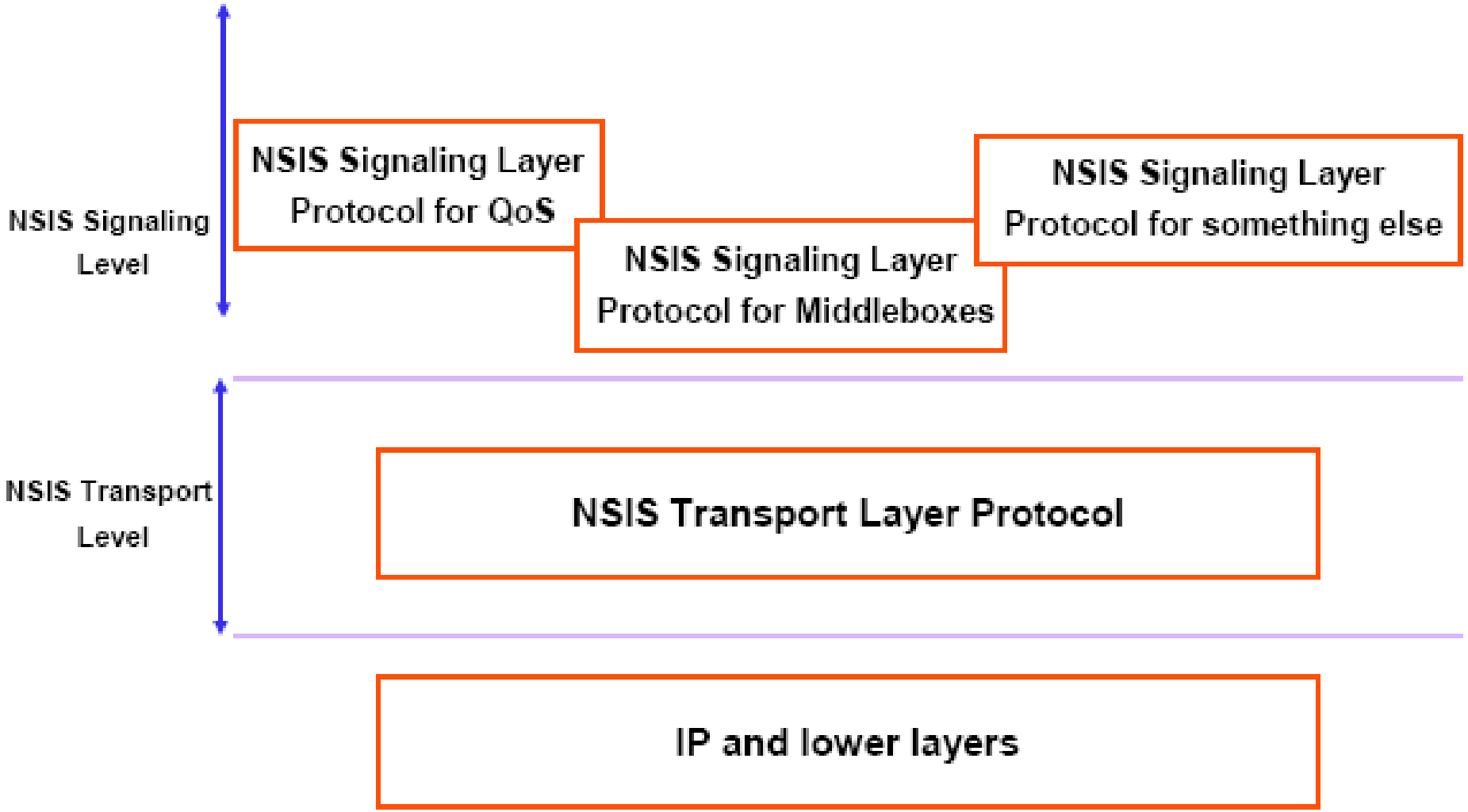
Main Conclusions Related to Aggregated RSVP (RFC 3175)

- Use a policy to maintain the amount of bandwidth required on a given aggregate reservation taking into account the sum of the underlying end-to-end reservations, while endeavoring to change it infrequently
- The number of the aggregated RSVP reservation states within a network will be significantly decreased but depends on:
  - The number of aggregators/deaggregators
  - The number of DiffServ Code Points (DSCPs) used
- Such solutions (policies) are very useful assuming that cost of the overprovisioned bandwidth is significant
  - In certain networks, where overprovisioning is not practical due to high costs of transmission links, a more dynamic QoS provisioning solution is needed

Next Steps in Signaling (NSIS)

- IETF NSIS WG
  - Chartered in November 2001
- Main goals
  - Signaling requirements
  - Analysis of existing signaling protocols
  - Signaling frameworks/protocols
- IP signaling protocol with QoS as first use case is the main goal, focusing on a two-layer signaling approach
- Reuse existing technologies wherever possible
- No development of new resource allocation protocol
- Use existing signaling (i. e., RSVP) as the basis

Framework: NSIS Protocol Components



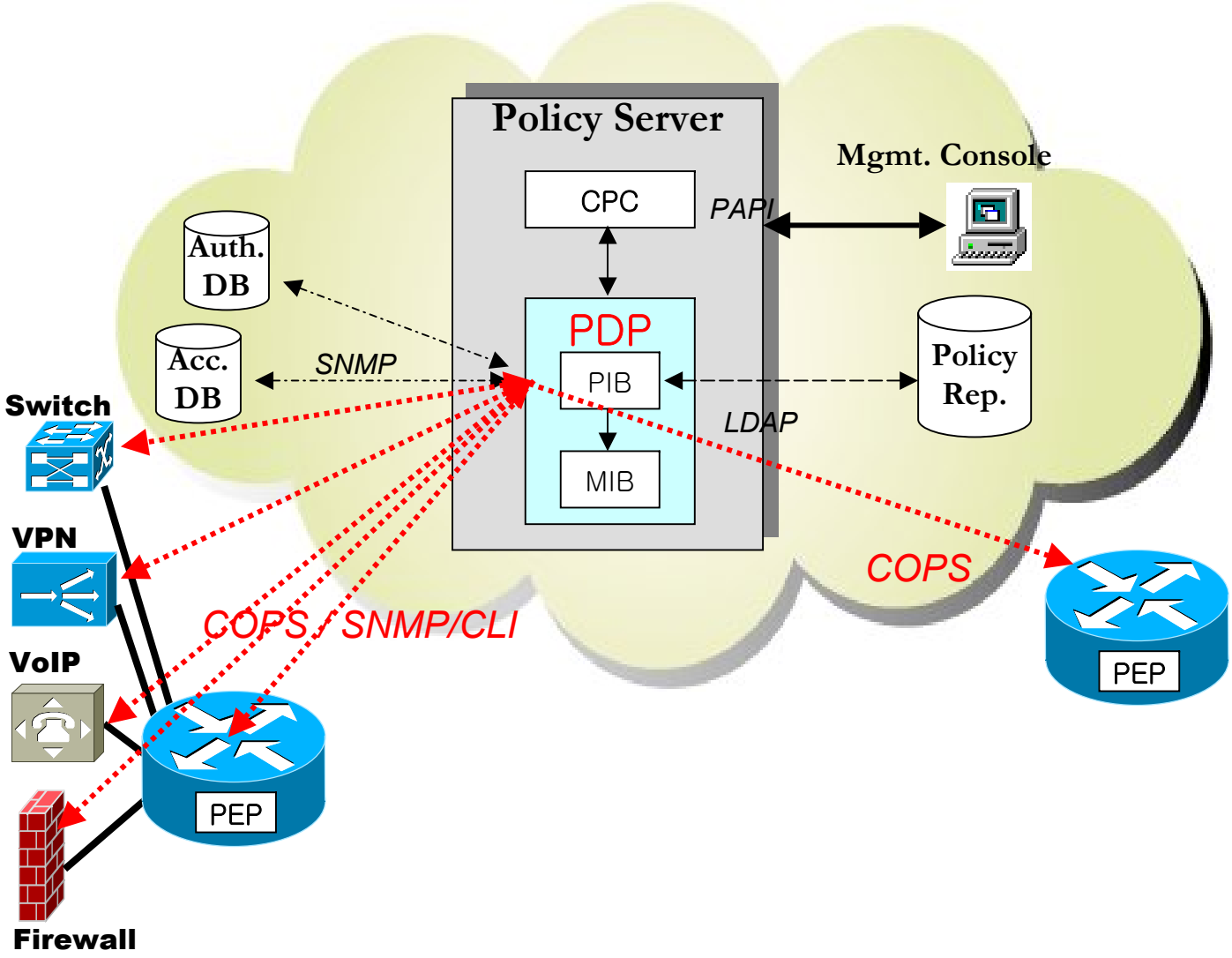
Policy Framework (1)

Why we need Policy?

- **Over-provisioning for QoS doesn't solve the problem????**
- **Complexity and heterogeneity make it harder to maintain network goals (manual labor won't work)**
- **Pervasive to virtually all modern net protocols:MPLS TE, DiffServ Provisioning, etc.**
- **DiffServ = Simple : Not really! => Provisioning E-2-E services out of PHBs is hard!**

- ❑ To define accurate use of resources
- ❑ To ensure end-to-end QoS commitment
- ❑ To allow centralized and consistent network policing
- ❑ To match business requirements

Policy Framework (2)



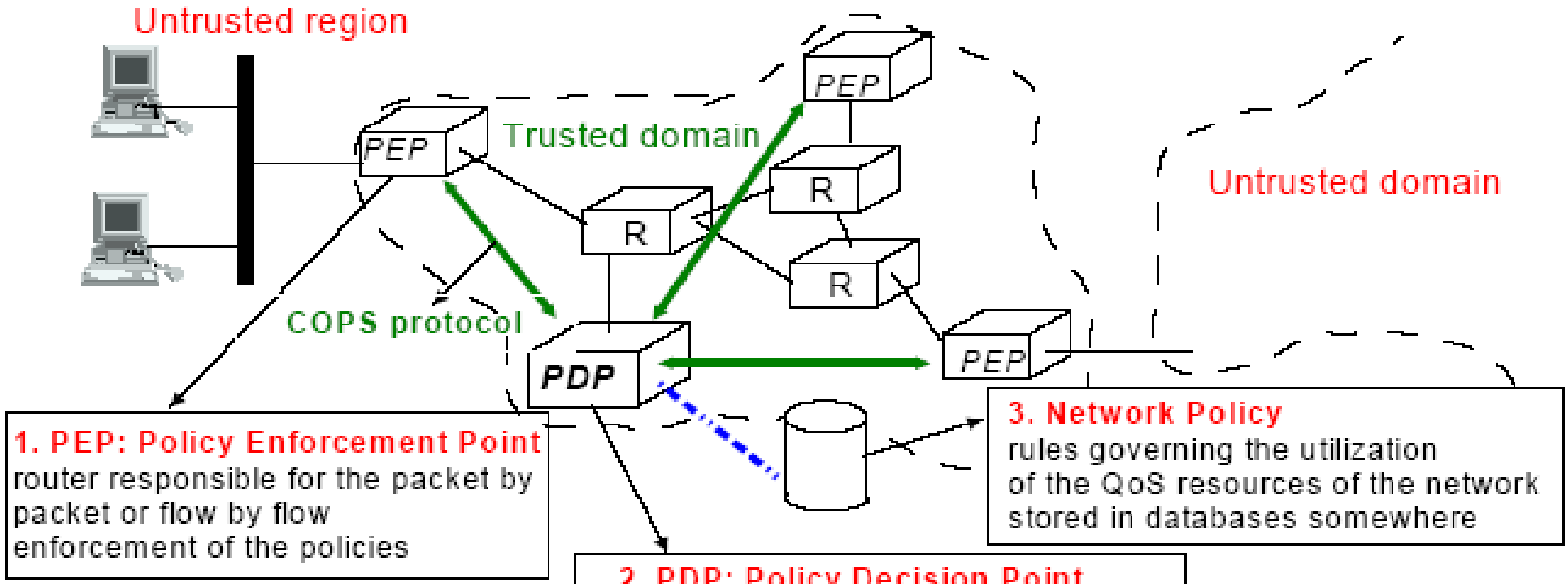
Policy Framework (3)

- Policy Framework Standardization in IETF
  - Architecture, terminology, building blocks
  - Last calls on Core and QoS Schema
- RAP (Resource Allocation Protocol)
  - COPS-Base [RFC2748]
  - Outsourcing (pull) model: COPS-RSVP [RFC 2749-2752, ...]
  - Provisioning (push) model: COPS-PR [RFC 3084, ...]
  - PIB (Policy Information Base) definition [RFC 2753, ...]
- SNMPConf
  - Less policy more configuration
  - Adopts the PIB model, attempts to use SNMP transport
- IPSP (IP Security Policy)

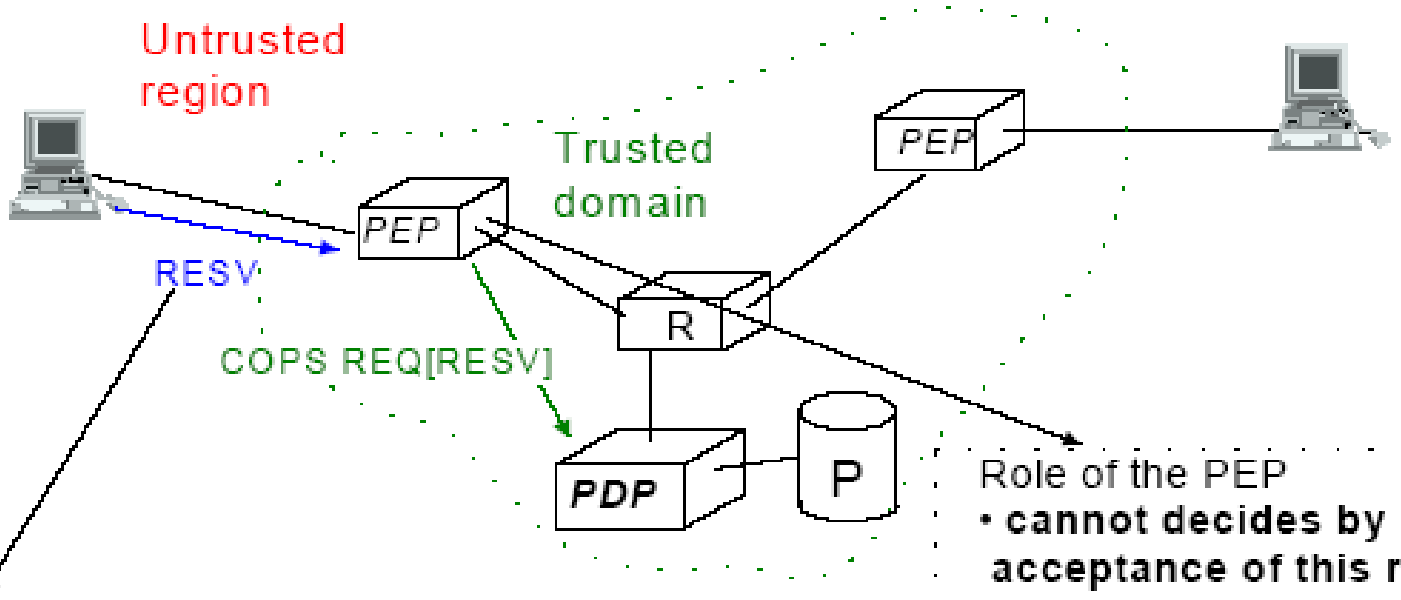
Policy Model

- Principles
  - policy applies to one domain and controls the operation of the entire domain
- The policy model is composed of three elements
  1. The Policy itself
     - defines how the network resources can be used
  2. The Policy Decision Point (PDP)
     - a centralized application that performs admission control based on the policy, the state of the network and the RSVP requests
  3. The Policy Enforcement Points (PEP)
     - the routers that are responsible for the actual enforcement of the decisions taken by the Policy Decision Point
  - The PEP and PDP communicate through a specific protocol : Common Open Policy Service (COPS)

Policy Model



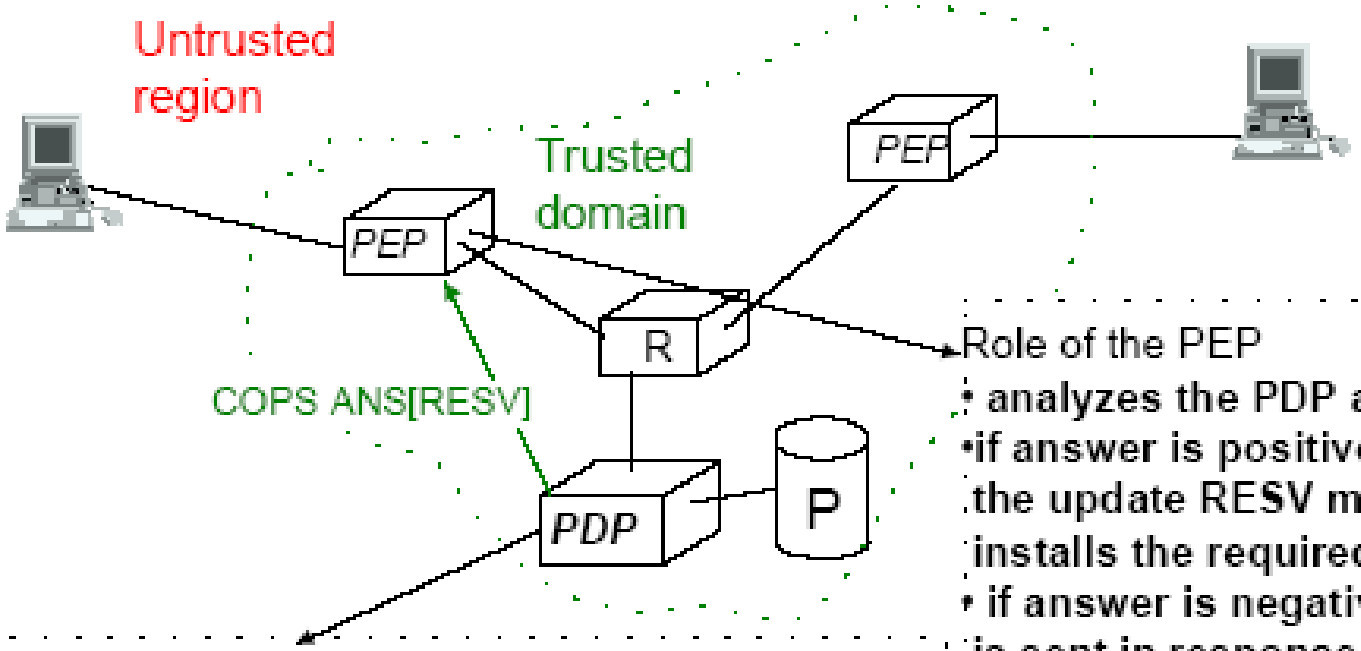
Policy Model in operation(1)



RESV message
- sender identification
- reservation
- application identification
- user identification
- optional signature for authentication

Role of the PEP
- cannot decides by itself the acceptance of this reservation
- extracts the relevant parameters of the RESV message
- sends contents of RESV to PDP through COPS protocol
- waits for PDP answer

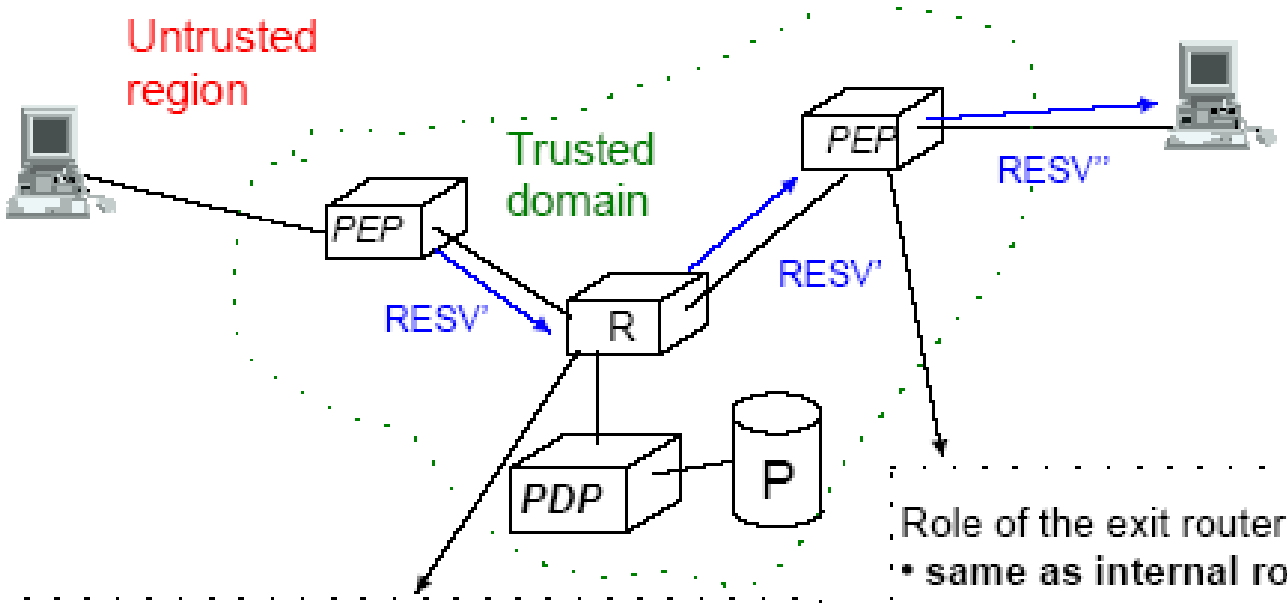
Policy Model in operation(2)



Role of the PEP
- analyzes the PDP answer
- if answer is positive, propagates the update RESV message and installs the required state in classifier
- if answer is negative, rejection is sent in response to RESV

Role of the PDP
- Receives the RESV message through COPS
- analyzes the request based on policy and current state of the network
- provides admission control decision to requesting PEP through COPS
- PDP answer, if positive, specifies the preemption and defending priorities to be used

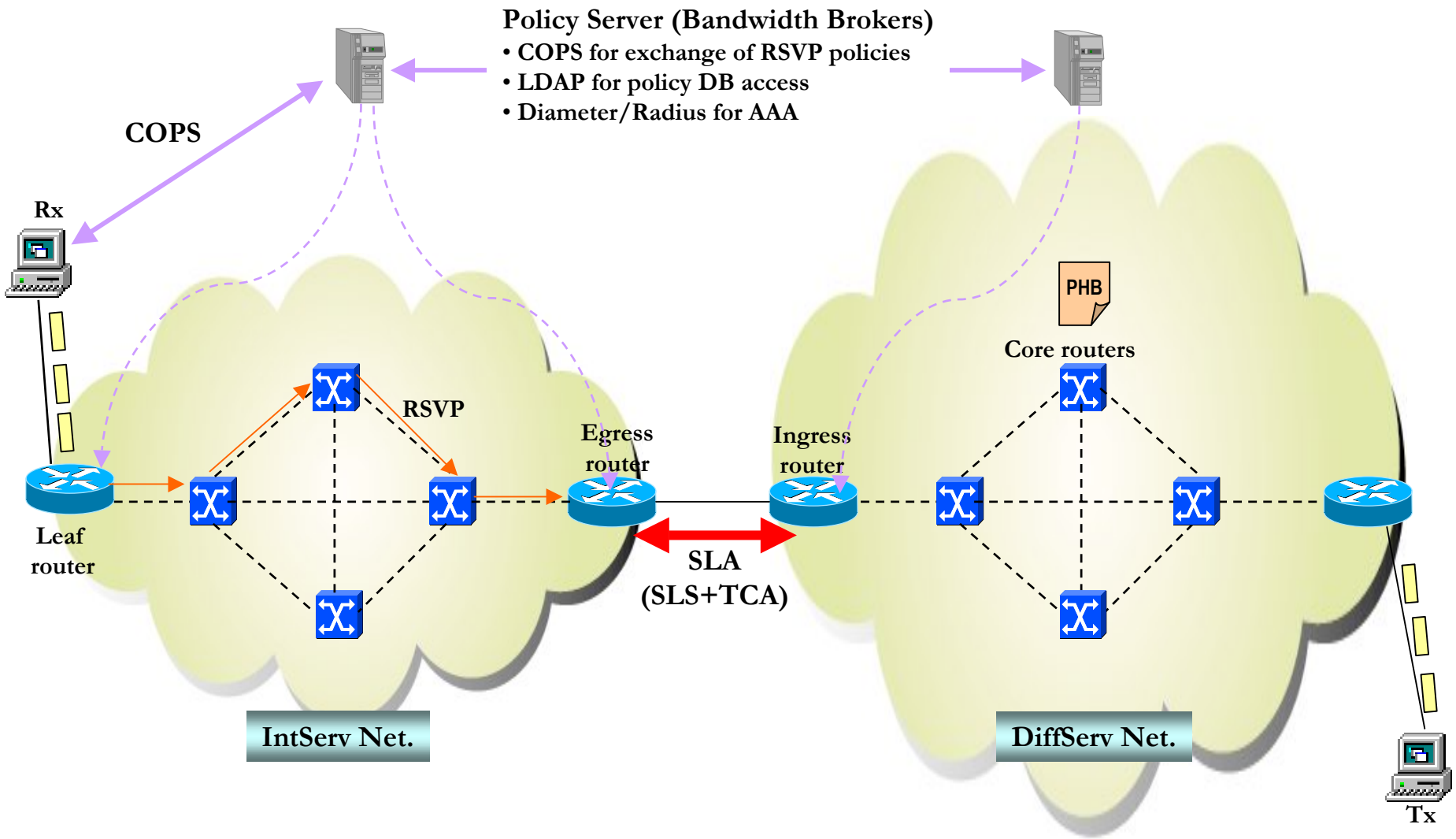
Policy Model in operation(3)



Role of the internal router
- knows that the RESV message has been accepted by PEP
- no interaction is required with PDP
- router performs basic admission control and possibly removes existing flows is resources are low and setup priority is high
- state for the flow is installed in router

Role of the exit router
- same as internal router plus
- exit router may remove some of the policy information in the RESV message (preemption and defending priorities)

Policy Framework (4)



Summary

- We review briefly :
  - Network Models
    - Integrated services, RVSP-TE
    - Differentiated services
    - MPLS & DiffServ-Aware MPLS
  - QoS Network Function Elements
    - Traffic classification, Traffic Conditioning, Queue managements, Schedulings
  - QoS Signaling (Management / Control Plane)